



© Copyright 1998 FASA Interactive Technologies, Inc.
MechWarrior is a trademark of FASA Corporation.
Confidential – For Internal Use Only – DO NOT DISTRIBUTE



Design Document

Version: 1.27

Created: 03/11/1998

Version Date: 9/1/1998

Author: David Luehmann

**©1998 FASA Interactive Technologies
All Rights Reserved**

CONFIDENTIAL: The information contained herein is confidential and a trade secret. It is unlawful to read, duplicate or distribute this information without the express written consent of the principle owner, FASA Interactive Technologies. Unless otherwise indicated by the principles, this document is consigned to the recipient for 30 days, after which time it must be returned to the principles or destroyed. By opening this document, the recipient is accepting the responsibility of adhering to these regulations.



Table of Contents

TABLE OF CONTENTS.....	2
OVERVIEW COMMENTS.....	3
PRODUCT PLATFORM.....	4
SIM VS. ACTION.....	6
THE INTEL CONNECTION.....	6
GAME CONCEPT.....	7
GAMEPLAY EXAMPLE.....	9
GAMEPLAY EXAMPLE II.....	10
TECHNICAL CHARACTERISTICS.....	11
THE GAME EXPERIENCE.....	13
MAJOR GAME ELEMENTS.....	16
‘MECHS.....	16
PILOTS.....	17
IN GAME SHELL.....	19
MENUS AND GAME FLOW.....	25
GAME CONTROLS / OPTIONS.....	31
‘MECH SUB-SYSTEMS.....	34
RESOURCE MANAGEMENT.....	56
NETWORKING.....	58
SCORING.....	62
MISSION TOOLS.....	62
AI SYSTEM.....	62



Overview Comments

‘MechWarrior IV will be the next generation of real time combat simulation games. It will establish the new standard in soft simulation gameplay, outstanding graphics and depth of story and missions.

Outstanding single-player gameplay is the primary focus of the development process. We are all aware of the growth and attraction of online gaming. One day these games may even become highly profitable. However, the largest section and dollar-producing elements in gaming is single-player games and that is our highest priority.

‘MechWarrior IV will greatly surpass its predecessors with fast, responsive controls, an intuitive interface, easy to use subsystems, compelling enemy artificial intelligence, tension-filled pacing and a healthy dose of challenging simulated action.

The game world created will be alive and dynamic. The player will be immersed in an interactive world that provides tactical and strategic benefits only to those who interact with it wisely. A ‘MechWarrior can utilize the environment and its objects to their advantage. For example, destroying a bridge will hinder the enemy, a building can provide cover and a rock formation can collapse onto an enemy in a trap. The player will *feel* like a part of the world and will experience what it is like to be in an enormous walking battle machine.

Fantastic visual effects sell games and eye candy shots bring great exposure. ‘MechWarrior IV will fill the screen with intense special effects during combat situations. Phosphorescent lasers will fill the area with intense color, streaming rockets will flare with blazing fire leaving smoking trails and buildings and vehicles will be consumed by massive fiery explosions, crumbling to smoldering rubble. Each individual mission will have details rather than diversity. Objects in each mission map will be fully articulated and will be intertwined with the world and not haphazardly strewn on top of it.

The in game experience will be backed up by details in the supporting features. The story and mission structure will follow a linear progression and will immerse the player, making him feel and believe he is a part of the universe and plays a critical role that will shape the future. Each mission will have depth and reason and will involve strategic elements that a real ‘MechWarrior would face on the field of battle. Resources must be managed and the player will covet rare and powerful equipment.



Product Platform

Living world with rich and detailed environment

The world is not empty.

The mission maps will be fully populated. We will do this by utilizing 3D technology advances to put more polygon objects into the world, not just more polygons into each object. This is not to say we will fill the world with a multitude of unattractive objects, rather we will use the power of new technologies to add depth to the game, not to create 30,000 polygon 'Mechs.

The world will have life other than just the player adding to the level of realism.

Non-player characters will have an agenda and will act upon that agenda. The world will not be an empty place that the player is just walking through.

1. Base personnel raise the flag in the morning.
2. Damaged vehicles will be towed into base hangers.
3. Other daily activities will take place.

The environment will evolve.

Major features in the game world will evolve based upon the player's actions. Mission flow will be designed so that players will return to areas previously visited to witness these changes.

1. When a bridge is destroyed, it will next appear in a state of partial repair with work still under way.
2. A broken dam will have flooded a large portion of the map.
3. A series of destroyed warehouses will be partially replaced.

Strategic elements during gameplay

Tactical Environment

Elements in each mission have tactical value and will influence the methods in which both the player and game AI's respond to situations.

1. A bridge is blown up and all non-jump capable 'Mechs must find a different route.
2. A sensor has been tripped leading enemy 'Mechs into an ambush.
3. Power to a radar system is cut, blinding the enemy defenses.
4. A building provides cover from enemy fire.

Resource Management

Resources within the game will be limited and strategic management of items and pilots will be critical.

1. A loss of an elite pilot or 'Mech will make future missions more difficult.
2. Choosing the right 'Mech, weapons and pilot will be critical to success.
3. Better AI levels will make wingmen an asset instead of a liability.
4. Capturing a supply depot will give the player new weapons or 'Mechs to work with.
5. Salvaged equipment will help fill the inventory coffers.



More immersive vehicles with superior motion dynamics.

Rough Terrain Navigation

BattleMechs are designed to go where normal fighting vehicles cannot tread over and through rough, difficult terrain. Intelligent animation will allow the 'Mechs to traverse over rocks, trees, scrub, small vehicles and buildings in a fast fluid motion.

Vehicle Dynamics

The 'Mechs will be fast moving and will accelerate and decelerate quickly. 'Mechs are capable of rapidly engaging target, head-on or by shooting 'over the shoulder.' Each BattleMech will also have its own unique set of motion characteristics that will distinguish it from the other 'Mech designs.

Virtual Cockpit

Each 'Mech will also have a 3D virtual cockpit which will move in relation to the 'Mech status. As a 'Mech begins to move and gain speed, the cockpit will bounce in step with the speed of the 'Mech, larger bounces at first and then smoothing out as the speed increases and the gyros begin to adjust to the motion. A damaged leg will provide a unique bob and the cage will rock when taking damage.

The Player will feel the size and power of a BattleMech

Scale

BattleMechs are big and we will use this size to our advantage. The Earthsiege and Heavy Gear products have small vehicles. BattleMechs can step on and crush Hercs and Gears. All of the elements in the game will be designed to reinforce the size of the BattleMech.

Audio Feedback

The audio effects will indicate a large heavy, object motion. The player will hear deep thundering footsteps, large machinery sounds and of course, booming explosions.

Atmosphere Feedback

Volumetric particle effects will provide the player with visual feedback as to the 'Mech speed. As the 'Mech begins to gather speed, dust caused by the 'Mech steps will zip around the windscreen demonstrating to the player the high rate of speed. Other visual speed effects will include rain, snow, fog and debris from explosions.



Sim vs. Action

There has been quite a level of discussion about how much action to put into 'MechWarrior 4 vs. how much simulator based realism should be in the game. In fact as I type this, the BadKarma web site is holding a fan vote on whether 'MechWarrior IV should be an action or simulation game.

'MechWarrior IV will continue the successful trend of tending toward more action than simulation. Heavy emphasis will be placed on visual and audible effects. However, at the same time will not ignore the simulation aspect of the game. Heat management, weapons recharge, the 'MechLab and other simulation components will still play critical roles in the game, but 'MechWarrior IV will not be a hardcore simulation game.

The Intel connection

The game will be designed specifically to run on the Intel Katmai processor. Our release will coincide with the second round of titles that are designed to support the Katmai technology during the holiday season of 1999.

The Katmai technology will give MW4 significant processing power designed specifically for advanced graphics and other visual and audio effects.

Other systems, such as the current 400Mhz Pentium 2's, will be supported as low-end systems. Meaning, aspects of the game will not be enabled on these machines. Specifically, features such as high poly count LOD objects, distant viewing volumes and special lighting effects, which are very processor intensive, may be reduced or in certain cases eliminated in order for the product to run on these low-end systems.

By specifically supporting this new processor technology, we expect significant support from Intel during the production of MW4. Intel will provide various marketing support programs, free development hardware systems and specific expert technical support.

This Intel connection will translate into greater market exposure, longer product shelf life and ultimately, ***significantly higher sales volume.***



Game Concept

'Mechwarrior IV is a real time, 3D, action-packed, vehicle-combat simulation game that takes place primarily in outdoor terrain systems. Based on the highly successful world of BattleTech, 'MechWarrior IV will expand and improve upon many of the successful features found in the previous 'MechWarrior products and add a few new ones.

In the game, the player takes the role of a 'MechWarrior who pilots a huge walking tank which can reach sizes of over 100 tons. Armed to the teeth with awesome weapons and technology, the player will set out to take control of a planet caught in civil war. But the player will not be alone. He will have support from a higher house who will provide him with 'MechWarrior wingmen, known as lancemates, and resources which he must manage and use to achieve a variety of challenges before reaching his overall objective.

The game takes place in the midst of the Steiner-Davion civil war. While Victor Davion has been off fighting the Clans, successfully driving the Smoke Jaguars from the Inner Sphere and crushing any possibility of a renewed Clan Invasion, his sister, Catherine (Katrina) Steiner has not been idle. Using her powers of propaganda to paint her brother as a ruthless despotic militarist, she has convinced much of the Federated Commonwealth that Victor's sole intention is turn over the nation to the Draconis Combine, their ancient and hated enemy. Seizing the opportunity to unite the Federated Commonwealth under her own leadership, she has dispatched her troops across the FC to take control of any worlds still loyal to Prince Victor.

'MechWarrior IV provides two separate, but related campaign games for the player to enjoy. Each campaign spans 15 missions and has an independent storyline and mission layout. The campaigns overlap in theme, since they each chronicle the fate of the system during the Steiner-Davion civil war, but from different perspectives. Since both campaigns take place at the same time and in the same system, extensive reuse of resources will be permitted and in each campaign the player will revisit battlefields and re-experience events that he has previously seen from the enemy's side.

In the Steiner campaign, the player takes the role of a member of the Lyran Armed Forces, dedicated to capturing and exploiting recalcitrant worlds for the greater good of a united Federated Commonwealth. The player must enter the system, establish a base of operations, eliminate the rebellious military forces loyal to Prince Victor and funnel the productive capacities of the system into the greater Steiner war effort. Along the way, it is expected that he will crush any civilian resistance and cure the populace of their misguided loyalties with the usual Lyran efficiency. The campaign begins with the player's attempt to conquer the system and ends with his successful elimination of the few remaining bandits and rebels that refuse to acknowledge the necessity of order.

In the Davion campaign, the player takes the role of the son of the local Duke, recently returned from campaigns against the Clans. Witnessing the invasion of his system by Steiner troops, he and his small band of fellow veterans vanish into hiding, using their hard won skills to aid the defense against the Lyran onslaught. When the regular army crumbles, his band continues a campaign of resistance aimed at embarrassing the Steiner commander and loosening his hold on the system. Eventually, the player battles the Steiner commander to restore himself as the rightful ruler of the planet.

The core game engine will be an evolution of internally developed technologies and will utilize a tool and shared code library set known as GameOS. The GameOS system allows the core development team a freedom from the issues presented by wide range PC architectures and Windows platforms. Through a layering system GameOS will provide 2D and 3D rendering, rasterizing, input devices, video playback, audio playback, file IO and networking API layers to the main game simulation code. All of these layers are designed to specifically support the Windows based operating systems.



The simulation engine will also be based upon internally created libraries known as Munga, Adept and Stuff. This game specific engine will track all of the simulation elements found within the game including heat generation and reduction systems, weapons systems, inventory and resource management, AI behaviors and more. We will identify and draw out specific simulation features from these libraries, such as missile trajectory algorithms, 3D targeting systems, etc. and incorporate these elements into the 'MechWarrior IV specific simulation engine when possible.

Most of the tools used to develop 'MechWarrior IV will be created during the pre-production phase of the project. Our intention is create a tool set that allows for a very clean and fast tool path from material creation to game integration making actual game development as simple as possible. These tools will support such materials as terrain systems, animations, visual effects, audio effects, AI routines, mission scripting and mission design and layouts. All of these elements can be quick tested "in game" by the creator before integration. These tools will reduce both overall iterations and the time between iterations.



Gameplay Example

The player receives the briefing for his next mission in his Mission Briefing Screen. The harsh voice of the Lyran Alliance Armed Forces Kommandant barks out instructions as the screen displays schematics of the operations area and its objectives. Although the player has routed the regular army and has the forces loyal to the Duke on the run, knots of Davion supporters still stubbornly resist. The current mission requires him to reconnoiter the rough desert canyons known as Hell's Hideout, which has become a staging ground for Davion loyalists in their raids on local mining operations. With determination, the player transports himself to the MechLab to outfit his team for the encounter.

In the MechLab, he begins by deciding which machines to take on this mission. LAAF command authorizes 120 tons of 'Mech for this endeavor and the player decides to split this between a 75 ton *Orion* and a 35 ton *Owens*, combining firepower and agility. Because the close confines of the canyons will restrict his encounter ranges, he customizes the *Orion* with a massive AC20 and four medium lasers, giving him lots of up close punch. He assigns the *Owens* to his favorite wingman, Lieutenant Goebbel, who will accompany him on the mission. Satisfied with his configuration choices, he enters the dropship's Mech bay and watches as the giant deployment doors swing open to reveal the rapidly approaching planet beneath him.

Entering the mission, the player curses his scanner readout. Heavy EM interference throughout the rocky terrain limits his sensor range to a few hundred meters, turning the desolate canyons into a garden of potential ambush sites. The crackling voice of an intelligence officer sounds on his com channel, informing him that a local transformer is the source of most of the interference; destroy that, and sensor function should return to normal. Rather than risk being swarmed by undetectable enemies, the player decides to take his informant's advice. He orders the *Owens* forward to flush out any resistance while he brings up the rear with his superior firepower.

The two 'Mechs crash easily through the sparse and dusty vegetation coating the canyon floor. The player pauses briefly to examine a precariously balanced rock formation, when he catches the flash of a missile launch off to his right. Goebbel's *Owens* staggers as it is hit in the side by an SRM and two fast hovercrafts race toward it, firing at its legs. Throttling his *Orion* into a run, the player trains his reticule on the lead hovercraft. The AC20 indicator on the reticule's targeting ring blinks into visibility, indicating that the hovercraft is in range of this awesome weapon. A quick trigger press sends a burst of heavy shells towards the vehicle, which explodes into spectacular flames as the projectile's impact. The *Owens* engages the remaining craft with its small lasers as the player spins his sensor-blinded 'Mech in an attempt to acquire new targets.

The missiles are being launched from a deep cave in the side of a nearby mesa, probably fired by a team or two of guerrilla fighters. Rambling towards it, the player shrugs off a few hits from the SRMs and levels all his weapons on the bunker, collapsing the cavern entrance with a few volleys of laser light. Goebbel's professional voice cuts in over the com line, requesting support and indicating heavy damage sustained. The player curses the EM interference, which keeps him from seeing whatever enemy is giving his comrade so much trouble. He rounds the mesa just in time to see the *Owen's* right leg vaporized by twin PPC bolts as the 'Mech bounces off the side of a rocky outcropping and clatters to the ground like a broken toy. When the smoke clears, the player sees the hulking specter of an 80 ton *Awesome* assault 'Mech bearing faded AFFC insignia. A last refugee from the defeated regular forces, no doubt, but deadly just the same. The player uses the EM distortion for cover as he ducks outside visible range behind the bulk of a massive sandstone boulder. Turning the corner, he continues parallel to the *Awesome* for another few hundred meters, dashing through gaps between mesas in case his opponent should be waiting for a clean shot.



“With luck,” he thinks, “that *Awesome* should have continued on his way to my last position, which would put him directly beneath that hanging rock formation”. Luck is with him, and his medium lasers blaze into the shaky base of the formation, sending it toppling into the rear of the unsuspecting enemy. Chunks of armor spall from the *Awesome* as it hunches forward, then rotates its torso and swings back an arm to level its left Particle Projector Cannon at the player’s ‘Mech. The HUD display de-resolves as the azure crackle of the PPC discharge washes over the player’s cockpit. A glance at the damage display shows heavy damage to his right front armor, but the *Awesome* fares worse – its rear torso glows a steady red in the display. Pushing his mech to the limit, the player loops his machine as widely as possible in the narrow confines of the tight canyon. Agile for a heavy ‘Mech, the *Orion* out-turns the cumbersome *Awesome* and the player once again has his reticule centered on the vulnerable rear. Two quick bursts of autocannon fire strip the remaining armor from the center rear torso before the player triggers all four medium lasers, sending a quintet of blue beams lancing into the vital engine and gyro of his opponent. The Davion pilot ejects, and the player watches his escape pod soar into the sky as the *Awesome* lands with a crash at his *Orion*’s feet. “That has got to be the toughest Mech those ragged resisters can field,” he thinks, as he lines up with his nav point and heads onward toward his objective.

Gameplay Example II

The player is watching a single renegade Thor running through a desert canyon. It’s sprinting at full speed, smashing through the small scrub and clumps of rock, a dust cloud marking its trail. It’s a perfect opportunity to get an easy kill. He targets the running ‘Mech, the PPC’s on his Atlas are well in range. Just a quick click of the trigger and two bright bolts of lightning flash, lancing out toward the enemy ‘Mech. The Thor is still about 700 meters out so a clean hit will be difficult. Luck today seems to be with the player, both of the PPC bolts hit the ‘Mechs’ right leg in mid stride. The damage is immediate and severe, the energy actually splashes around the Leg. The leg is ruined, its’ armor must have already been damaged, what was left was melted by the energy bolt, its actuators then melt and seize from the heat, it going to be ugly.

The Pilot can’t do anything but hold on tight. The Thor was traveling at just over 80 KPH when it’s leg was gimped, and almost in slow motion the player witnesses it topple forward crashing onto its chest. This display is far from over, the now prone ‘Mech is still traveling at a good 70 KPH, plowing forward through the terrain, carving a trench in the hard cracked dirt. Debris is splashing off the front of the ‘Mech, it reminiscent of a plane crash. Armor, a ‘Mech arm, small trees, scrub, rock and dirt blast twenty feet from the still moving mass of metal. A trail of fire ignites about 10 meters behind the now slowing ‘Mech. Must have had missiles on that ‘Mech whose fuel is now igniting. As the ‘Mech stops in a cloud of smoke and dust, the fire trail catches up to it, a few ammo explosions rock the ‘Mech and player can hear the devastation even from this distance.

The player throttles up moving toward the downed ‘Mech, it shouldn’t be hard to finish the ‘Mech off, if somehow managed to survive.



Technical Characteristics

OS

'MechWarrior IV is designed to run within the Windows98 operating system. We will also be compatible with other Windows systems, which meets our DirectX requirements.

Game OS

MW4 will be the first of many games FASA will produce that use the GameOS system exclusively. The GameOS system is a layered technology piece, which allows game specific mechanics to interface easily with other drivers such as audio, 2D / 3D rendering, interface, networking and video systems.

Resolution

The interface, in game screens and HUD will be designed to operate at 800 x 600 resolutions. We will consider supporting higher and lower resolutions as the market may demand.

Rendering

The target for the rendering engine is 6,000 polygons per frame at 30 frames per second. 3D hardware support will be required and Katmai Pentium processors will be recommended. The rendering engine will support a host of graphic technologies including; numerous dynamic lights, alpha channels, mip-mapping, LOD fading, additive blending, texture blending, environment mapping, gourad shading, lightmaps, vertex fogging, vertex coloring, run time triangle tessellation, particle effects (light particles and particle collisions), bullet holes (cards), footprints (cards), animated textures, a multi-layered sky (infinite render), horizon circle lines (infinite render), post-processed frame buffer effects (heat wave, etc.), dynamic shadows and bump-mapping.

Networking

'MechWarrior IV will use a hybrid client/server lobby based meeting system to support a wide variety of network connections. At least 8 players will be able to compete in a number of competitive match play types.

Audio

Sound effects will be presented to the player in 3D positional locations on our target hardware. The system will provide 3D positioning, panning and fading and other real-time filtering designed to bring a new level of realism to the simulation. As a player moves, sounds coming from the objects will appear to the player as if they are coming from the right, left or behind the player and increase or decrease in volume based on the distance from the player. Other real-time filtering will come in the form of echos, bouncing off large objects or muffled sounds, if an object is between the player and the sound source.

We intend to support accelerated 3D audio chips, such as the Aureal 3D chipset, if they are supported by DirectX7, DirectSound / DirectSound3D library via the GameOS layer.



FASA INTERACTIVE

© Copyright 1998 FASA Interactive Technologies, Inc.
MechWarrior is a trademark of FASA Corporation.
Confidential – For Internal Use Only – DO NOT DISTRIBUTE

Music

Music scores will be presented via a digital audio playback system, rather than midi or other dynamic music creation systems. This will ensure the best quality control over the music. Music will be used to set the pace at specific times during the game. When a player is moving through what appears to be an enemy free swamp, the music is menacing. When combat begins, the music reflects the adrenaline rush.

Media Type

Our primary delivery medium will be a single DVD ROM. Only one ROM will be required for local networking games of up to eight players. In all other instances, each player must have their own copy to play.

Target Platform

Intel Katmai class processor with an AGP Bus, 128 Megs of RAM, Accelerated 3D sound and polygon hardware.



The Game Experience

‘MechWarrior IV’ expands upon the basic science fiction combat simulation experience by involving the world as a tactile and strategic element. This is further expanded upon by a resource management system which offers the player a deeper level of immersiveness in the game.

Critical Game Features

The internal technology and experience of the company gives us a dramatic advantage over other games in the genre. We will use this advantage to provide both greater visual effects and increased playability to the simulation experience in the game.

‘Mechs

The player will have 16 ‘Mechs to command and fight against. Each will have its own unique look, animations and simulation features that distinguish one from another. The player will be able to both see and feel the unique personality of each of these awesome futuristic military machines.

Customizing

The player will be able to completely customize both the look and hardware systems in each ‘Mech via the ‘Mech Lab. The player will also be able to control the ‘Mech of each of his lancemates, matching each pilot’s skill with specific ‘Mech types.

Tools

With content creation tools in the box, the player will be able to create his own unique mission of the FASA maps. Through online trading, a huge number of missions will provide endless gameplay.

Combat Simulation

The combat simulation will support all the major weapon types found in the universe. Each will have its own particular destructive and visual effects. All the major sub-systems found within the ‘Mech will be supported including: weapons, sensors, damage, armor, collisions, movements and engine outputs.

Special Effects

‘MechWarrior IV will set a new standard in simulation effects. Gratuitous use of particles, lights and animated geometry will be used in each mission. Buildings will collapse as they explode and armor and limbs will fly from damaged ‘Mechs, leaving only fire and smoke behind.

Environments

The terrain created for the ‘MechWarrior IV project will be the most visually stunning and interactive terrain system created for a PC based game to date. In house rendering technology combined with the latest Intel hardware and expertise will enable us to create an advanced terrain system that can currently only be supported by specialized platforms.



Features will include high resolution texturing, shadows, superior lighting and terrain deformation. These features combined with exhilarating terrain design will create a world that is striking and strategically challenging. The terrain will be designed to present the player with the challenge of determining how to best use the terrain to accomplish their objectives.

Missions

The player will traverse 30 individual missions in two separate campaigns, each campaign providing a unique perspective of the world and its events. The player can then choose what types of hardware and pilots will be joining him in each mission. We will also provide a few additional maps specifically designed for multiplayer scenarios.

In Game Objects

The objects used in the 'MechWarrior IV environment will be the most true to life 3D models ever used in a real-time PC games. They will have higher polygon representation than has ever been available on the PC platform. Objects will have high resolution texturing, multiple levels of detail, animated textures, lighting and be articulated to represent conditions such as being destroyed, damaged, being rebuilt and moving. Players will be able to dynamically interact with the objects in the world using them for cover, to set ambushes or simply to satisfy their need to destroy.

Campaigns

A two-campaign system has been created for the player to traverse. We have designed a two campaign system for a few reasons. First, it will offer the player two different levels of difficulty and not limit a player who has become stuck on a single mission. Second, it also reduces the overall amount of resources that will need to be developed. Third, it allows for a greater level of detail in both design and execution of each mission area.

Resource Management / Inventory

A resource and inventory system will allow the player to choose what type of hardware to bring or save for further threats. Salvage will allow the player to add to the inventory coffers through the taking of enemy equipment that is captured or killed during combat.

Control Points

Control points are used as a high level scoring system, which the player is not directly aware of. A highly successful player will be given control points, which will allow him to learn of other potential tactical information or beneficial tertiary objectives.

AI

The artificial intelligence created for the 'MechWarrior IV project will provide the player compelling interaction with friendly and enemy forces. Artificial intelligence will be provided in various forms. This will mainly take the form of 'Mech pilot AI, but will also include pilot AI for other types of crafts, along with intelligent structures, such as a communications tower that will call for help if being attacked.



FASA INTERACTIVE

© Copyright 1998 FASA Interactive Technologies, Inc.
MechWarrior is a trademark of FASA Corporation.
Confidential – For Internal Use Only – DO NOT DISTRIBUTE

The artificial intelligence for the ‘Mech pilots will determine characteristics such as how well a ‘Mech can target, what attack pattern a ‘Mech will use, how well resources are utilized and how weapons are used. Pilots of various levels can be created to provide a dynamic adventure where no one type of ‘Mech always behaves the same.

As the game progresses, the pilots that accompany the ‘MechWarrior into battle will grow more skillful as their experience levels increase. They will improve in areas such as targeting, weapons use, attack patterns, use of ECM’s (electronic counter measures) and sub-system utilization. Experienced pilots will become valuable assets for completing advanced missions and achieving final victory.

The abilities of enemy forces will be determined by the mission designers. Overall intelligence attributes will be settable, such as smart pilots verses stupid pilots, along with more specific attributes such as levels of aggressiveness, weapon usage, sub-system management and primary targets. The enemy forces will have settable levels of artificial intelligence so that levels may be better scripted, allowing designers better control of the game flow and thus control the player’s experience. The designers, by controlling the abilities of the AI, will determine the tone and mood of the player’s experience at different points during the missions.

Networking

‘MechWarrior IV will support up to eight – sixteen players in death-match based combat games. ‘MechWarrior will offer full support of team vs. team games and will track online statistics and rank the top 1000 players in the online gaming world.



Major Game Elements

‘Mechs

Meticulous detail will be placed on the development of each individual ‘Mech. ‘Mech creation is one of the many areas where ‘MechWarrior IV will stand apart from the competition. Painstaking detail will be placed on the modeling and texturing of each ‘Mech and each will be animated to demonstrate the awesome size and power of the BattleMech.

Our plan on ‘Mech designs has changed numerous times and has hopefully now been settled. We are planning to create a series of 6 new ‘Mech for the product as well as including a mixture of the famous Mechs that have become the icons of the property. The new ‘Mechs created will be generated first visually by our internal team, once a basic look and feel has been developed the MW4 team will begin working with Fasa Publishing in order to maintain balance and create fictional background for each new Mech type.

Classic Mechs:

Clan:

Madcat
Thor
Novacat
Shadowcat
Uller

I.S.:

Atlas
Zeus
Raven
Owens
Catapult

New Mechs:

Currently under design.

Detailed states will be created to indicate the status of the ‘Mech during combat. As a ‘Mech incurs damage, its textures will be “burnt” through either vertex coloration or complete texture swapping. If a limb is destroyed, a smoking, sparking stump will remain.

The ‘Mechs will also be completely customizable both visually through custom textures and technically through weapons, armor and engine types, giving the player complete control over the vehicle in the simulation.

The ‘Mechs will also remain true to simulation guidelines. Unique characteristics such as turning radius, acceleration, critical spaces and torso twists will distinguish each ‘Mech. Heat management, sensor systems, generator output, weapons and targeting systems will also play critical roles in the game.



Pilots

The player does not fight alone in 'MechWarrior IV. In addition to himself, he can command up to four additional AI based 'MechWarriors from a pool of up to 16. Each AI warrior is treated much like a persistent resource, gaining experience and talent as the game progresses. Experienced pilots are highly valuable and losing one in combat can have heavy repercussions to the player. He may no longer have a powerful friend to rely on.

Each pilot will gain weapons skills based upon experience gained in combat. Further, each pilot will have a unique attitude that will dictate how he (or she) handles combat situations. A cocky, aggressive pilot may love close combat while a calm, cool-handed pilot may prefer longer-range engagements.

In combat, these pilot will speak to the player via digital audio, informing the player of the situation they are witnessing and replying to basic orders such as, "Yes sir!, Enemy Destroyed", etc., which will also reinforce to the player the unique character traits in each pilot.

The player when deploying will have the ability to choose which 'Mech and pilots they bring along in each individual mission.

Commanding Pilots

The player can send a series of commands to his lancemates during the simulation. This system will be broken down into a verb / noun system. Verbs will remain constant, nouns will change based on objects available in the current mission. Example nouns are enemy 'Mechs, Nav Locations, Buildings, Me (the player), etc.

It is important to note that the AI pilots will have 'normal', non-ordered behaviors that they will always follow. However, when the player wants a lancemate to follow out a specific order, they do so within this system, over-riding the basic AI behavior.

When giving an order, the player will choose which lancemates should receive the order, select the specific order, then select the specific noun.

Pilots will always respond to each command and attempt to carry out the orders if possible, even if it means certain death.

Commands

Engage

Causes a pilot to engage a specific enemy (noun) or to engage any enemy encountered. The battle will continue until one of the 'Mechs is destroyed or the order is changed.

Guard

Causes a pilot to guard an object until all threats are destroyed, the object being guarded is destroyed or the order is changed.



Cover Me

Causes a pilot to defend the player until either the player is killed or the order is changed.

Join Formation

Recalls a pilot from a previous order or location to join up with the group.

Goto

Sends a pilot to a specific noun location. Once the pilot arrives at the location, he will go into an idle mode and wait for the next command. If the pilot is not able to solve the path location, the command will auto-cancel.

Shutdown

Causes a pilot to shut down his 'Mech used for stealth or ambush missions. The pilot will remain in this mode until ordered otherwise or he is shot by an enemy 'Mech.

Recon

This is a fairly complicated command and if it proves too difficult to implement, it will be dropped from the command list.

A recon command will send a pilot to a specific location. Once at the recon location, the pilot will either succeed or fail based on whether or not he was discovered performing a 'walk around' the location.

If successful, the pilot will report back to the player if an enemy has been discovered or if the area is clear. If the pilot fails, he will report back that he has been discovered and will auto-engage the enemy unless ordered to do otherwise.



In Game Shell

The in game shell will contain three primary elements - the Briefing and Deployment area, the 'Mech Lab, and the Commander Log. Each of these elements will be provided to the player via a 2D graphical menus, which are overlaid on a 2D background prepared to look like a 3D environment.

The basic shell system is designed to further the player's feelings of being in the game. The main view will appear as the hanger bay of a dropship for the Davions, and a command HQ for the Steiners. In each campaign, the view will appear to filled with activitiy, 'Mechs will be moving about, techs will be performing repairs, life will be present and will be following an agenda.

When 2D interface menus are needed, a PDA will integrated into the scene. When not in use it will reside folded out of the way in the corner of the screen, when accessed, the PDA will flip out providing the player with all the menu and sub-menu options.

When a player chooses to deploy, he will return to the 'Mech Bay (or 'Mech Hanger) view, his 'Mech will be picked up by a large deployment claw and moved to a hanger bay door. Claxons will sound, lights will flash and the player will be dropped out of the ship onto the actual mission play area and start the simulation as seamlessly as possible.

Briefing / Deployment System

This section will provide the player with the overview of the mission briefing, a tactical map overview of the mission play area, a list of the objectives in the mission, a mission parameters area, a ready area and a force group area.

Briefing Area

The mission briefing window will contain a text list of the mission briefing and a tactical map of the mission area.

The briefing info will provide the fictional rational behind the mission also via text and also by a digital audio voice over from his commander.

The tactical map will give the player a high concept overview of the primary terrain elements and zones within the mission area. Included in this overview map will be the locations (if known) of the primary and secondary objectives, nav points and insertion and dust-off locations.

During the mission briefing, the navigation points and primary objective points will be plotted out by the commander making the player feel like they are involved in the planning process. (Only primary objectives are required for mission success.)

Objective Area

The objective window will simply list the specific objectives involved in the mission. The player must complete the primary objective to win the level. Tertiary objectives will be listed based upon the control points that the player has acquired during previous missions.



Mission Parameters

These specific mission parameters including maximum deployment weight, zone information and other specific information about the mission.

Maximum drop weight in tons will help the player choose what type of ‘Mechs or other equipment to bring.

Zone information will describe the basic terrain location to the player - a rocky desolate desert or frozen snow covered plains.

Other information such as the time of day and current weather conditions will also be provided in this section.

Ready Area

The ready area will depict each individual vehicle that is ready to deploy along with a small area that breaks down the pilot name and vehicle info.

The only real requirement to be considered “ready” is that a vehicle must have a pilot. The game will always suggest both pilots and vehicles for each mission. However, the player will have broad control over what and whom he chooses to bring and can customize both from within the ‘MechLab.

Below the ready area is a simple customize button. Clicking on this will bring the player to the ‘MechLab.

The player and his ‘Mech will always be listed first in the ready area.

Force Group

The force group is where the player lists which ‘MechWarriors, if any, he will bring along on the mission. To add or remove ready vehicles to this area, the player can just left click and drag the icon from the ready area to the force group area. The force group must be less than the maximum mission deployment weight. If the player attempts to exceed this weight restriction, the game will beep and inform the player of the problem.

Launch Button

A simple launch button will bring the player back to the Mech Bay view and execute the mission drop.

‘MechLab System

The ‘MechLab is a highly complex item which we’ve attempted to design with ease of use and understandability in mind. However certain interface issues will not be addressed until the pre-production phase of the project.

Functionality



The lab will allow a player to customize any 'Mech both visually and functionally by adding or removing weapons and other critical systems within a few certain rules namely, required systems, critical spaces, maximum chassis weight and lastly, weapons hard-points a new element to the 'MechWarrior series.

Weight

Each chassis has a maximum allowable weight which the 'Mech may not exceed. Every item in the game has a weight value associated with it. When outfitting a 'Mech, the player must adhere to this maximum weight value.

Items that that if added will put the 'Mech over the maximum weight of the chassis type will be grayed out.

Required Systems

Every 'Mech will require a few systems to operate. For gameplay reasons, these systems will include engines, gyros, sensors, a minimum number of heat sinks, and a minimum amount of armor.

The game will not allow the player to remove these required systems.

Critical Spaces

Each 'Mech chassis also allows for a unique number of critical spaces or open area, where each component of the 'Mech is carried. These spaces are distributed throughout each sub-section of the 'Mech, namely the torso, head, legs and arms.

Each item in the game also has a certain volume associated with it via critical spaces. (A PPC is 4 critical spaces in size.) When outfitting a 'Mech, the player cannot place an item that requires 4 critical spaces on an arm that has only 3 remaining spaces.

Items that require more free critical spaces than the current subsection allows for will be grayed out.

Hard Points

Unique to the 'MechWarrior series, the lab will require that the player places weapons in areas that are fitted for a specific weapon type.

For example, the Mad Cat clearly has two and only two missile packs, one on the left side of the torso, one on the right side of the torso. The player will not be able to place a PPC or other energy-based weapons on this hard-point. Only missile weapons can be placed here, any type or salvo, but only missiles. Each 'Mech chassis will be very unique in this fashion. This hard-point system will only apply to weapons, other equipment can be placed at will.

Only weapons that are allowed in the current 'Mech subsection will be available, other weapons will be grayed out.



Adding Items

To add an item to a 'Mech, the player can click and drag an item from the inventory window and drag and drop the item onto the image of the 'Mech in the 'Mech Bay view.

Move or Removing Items

To move an item from the 'Mech, the player must select the item from the critical location window and either drag and drop it onto a different critical location within the 'MechBay view or remove it from the 'Mech by dropping it in the inventory window.

Sections

The lab is divided into six sections in order to provide the player with an organized system that will minimize the learning curve involved in outfitting a 'Mech and allowing for rapid customization of each individual 'Mech.

Chassis

The chassis section will display information about the current 'Mech or vehicle-type including name, class, top speed, maximum tonnage and current pilot info (if any). The player will also be able to scroll through all the available chassis' sorted by vehicle type in ascending in weight.

Critical Location

Information about the critical spaces of the currently selected sub-section of the 'Mech is displayed here. Each critical space is displayed via a text description of the item contained in the critical space. Free spaces will be indicated with a dash. (Exactly as in 'MechWarrior 2 - Mercenaries)

Inventory

The inventory system can get quite complex as it deals with a large variety of item types. To address this, each item is categorized through a series of identifying tabs and a scroll bar so that a large number of items in a category can be displayed.

Missiles

This section will contain only missiles and will be sorted by range, then by type, then by salvo size. Each missile pack will come with one free ton of ammo!

Energy

Contains PPC's and Lasers sorted by type then range.



Projectiles

Machine Guns, Gauss Rifles and Autocannons and Artillery are found here. Also sorted by type and then range.

Ammo

Additional ammo for Missile and Projectile weapons will be found here.

Equipment

Non-weapons based equipment will be found here. This will include items such as Sensors, CASE, MASC, C3 computers, NARC Beacons and Armor.

With special items such as CASE or Special Armor types, the inventory window will change to a specific interface used to handle specific item allocation methods.

Groups

The player can setup specific weapon grouping configurations here. Each weapon on the 'Mech will be listed in text form. To change a weapon group setting the player will only have to click on the item to cycle through each group.

Paint

Specific visuals can be edited here including camouflage skin type and team logo.

Each 'Mech will have a few camo types designed for stealthiness in particular environments (jungle, swamp, etc.). We will also allow the player to select a custom created camo type. Custom skins will be loaded in bitmap form and will have to follow fairly rigid technical guidelines to work properly.

To adjust the skin that a 'Mech uses the player can select a new skin name from a simple drop down list. To create a new skin the player can load the texture files that are used by game (a series of tga's) perform any changes they would like. They must then save the file out under a different name which will then be available in the aforementioned drop down menu list.

Mostly used for multiplayer modes, the player can also choose a team icon which will be placed in predetermined locations on each 'Mech type. We will provide a few icons for the player to choose from or allow them to import custom created versions in bitmap form.



Pilots

Both active and inactive pilots will be listed in here along with specific details about the pilot's experience (regular, veteran, elite) along with personality traits and short fictional anecdotes about the pilot.

Lance Roster

Icon thumbnails of ACTIVE 'MechWarrior Pilots are displayed here. If the 'MechWarrior has been assigned to a particular 'Mech, a mini icon of the 'Mech will also be displayed next to the pilot icon.

Clicking on the player icon will select the 'Mech he is assigned to as the current 'Mech in the Lab.

Detail Info

The detail window will provide additional information regarding the item currently under the mouse pointer. This info will change based on the type. For a weapon, it will display the weapon name, inventory count, how much heat it generates, its effective range, its weight, recharge time and size.

'Mech Bay View

This window will show a high detail, real 3D version of the current chassis type, which can be snap rotated from front to rear views. The player may select the active sub-section by clicking on the actual 3D object. When a section is selected it will glow.

Commanders Log System

The Commanders Log provides the player with summary information about the last mission and also previously played missions.

Mission Recap

At the end of a mission, the player will be provided with what is essentially a high level summary of the player's performance. The status of mission objectives (completed / failed) will be displayed and if the player is located and succeeded in dealing with tertiary objectives.

Mission Statistics

This is a detailed sub-section of the mission recap. Additional information about the player's performance will be listed here including:

- The number of enemy 'Mech killed.
- The number of enemy structures killed or captured.
- The number of enemy vehicles killed or captured.
- The players proficiency with weapons.
- Salvage taken during the mission.
- Losses incurred during the mission.



FASA INTERACTIVE

© Copyright 1998 FASA Interactive Technologies, Inc.
MechWarrior is a trademark of FASA Corporation.
Confidential – For Internal Use Only – DO NOT DISTRIBUTE

Each element of these statistics will be tracked, given a point value and added (or subtracted) to give the player an overall experience number gained from the mission. This experience number will become the player's score. Then, at predetermined point levels, the player will receive rank promotion.

These total experience points can also be used to compare game skill levels, essentially bragging rights.

To promote these bragging rights, mission recap and mission summaries will be able to be printed to hard copy or saved as text files that can be posted on player web or other promotional web pages.

Commander Logs

The Commanders Log will allow the player to return to previously played missions and view the recap and statistics from previously played missions. A very simple VCR element will allow the player to view previous game highlights, much like the Tesla system. These highlight reels can then be shared and traded with other players online.

Next / Replay

Two last simple buttons will allow the player to continue on to the next mission in the campaign or replay the mission.

Menus and Game Flow

The Menus and game flow in Mechwarrior 4 has been designed for easy of use, natural flow and familiarity with its predecessors. While some of the particulars will be improved, such as the MechLab, the basic menus systems should be comfortable to our historical users.

Installation / Introduction

The first time the player runs the game after the initial installation the introduction FMV will automatically play. If the game has been played one or more times the player will automatically be taken to the main menu.

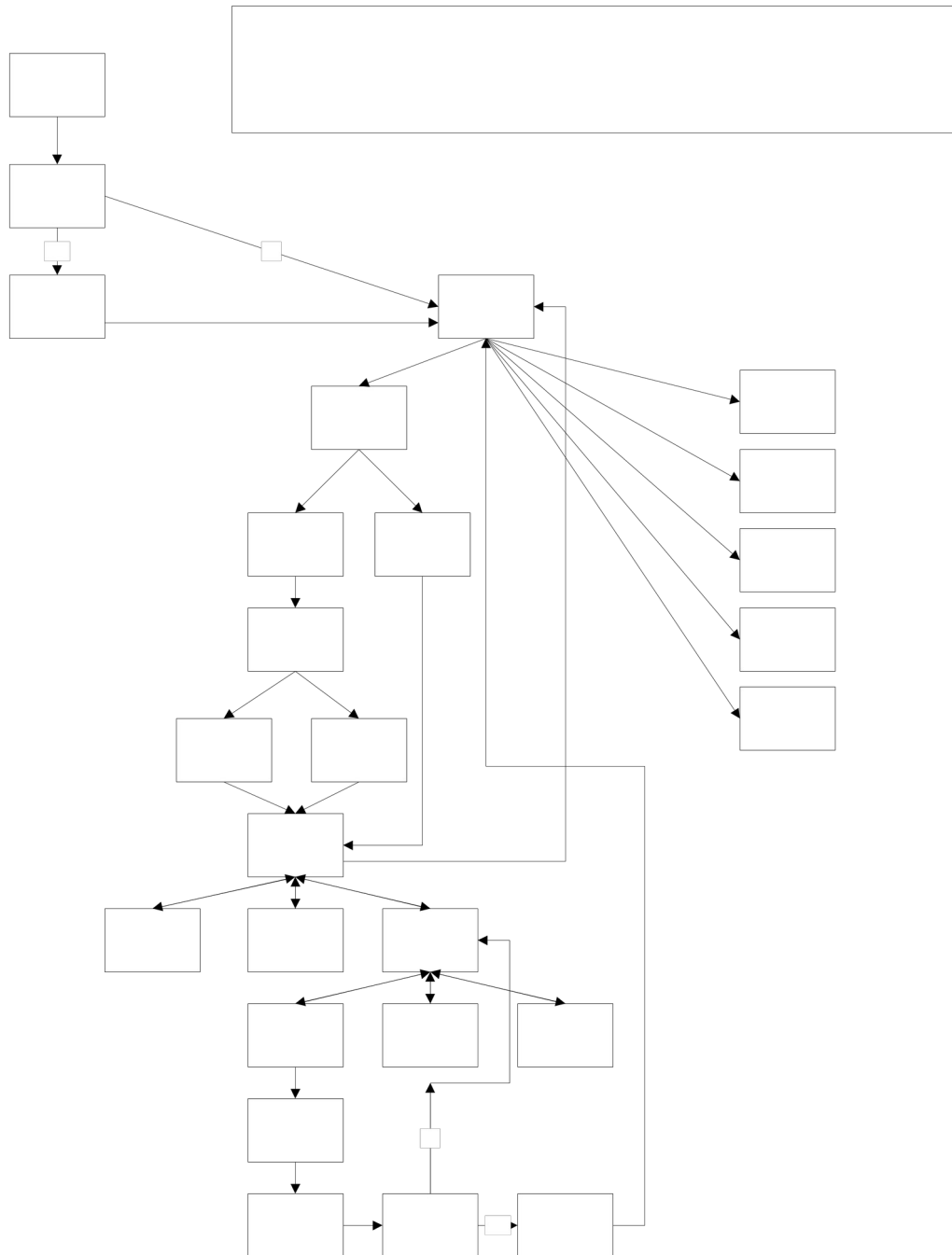
Main Menu

After the introduction animation plays, the player will reach the main game menu. This is the high level game menu, a conduit which will lead the player to a series of individual submenus and main game type menus.



Single Player

Single Player Flow Chart





Load

Selecting load, will open a dialog window that will list previously saved games (up to 8), selecting and loading saved game will return to player to last unfinished mission in the simulation (players cannot save mid-mission), at the main game menu.

New

Selecting a new single player game will bring the player to an alliance screen much like that in Mechwarrior 2. The player will have to select to align themselves with the Steiner or Davion faction. Once selected, a short FMV will play that provides a short narrative story background for the chosen faction.

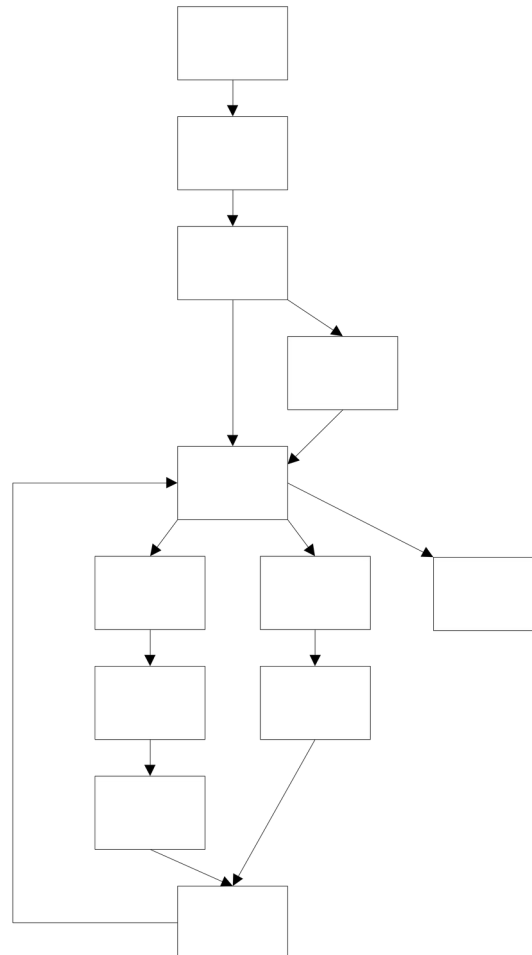
After the FMV has played, the player will be presented with the main game menu where they can adjust options or go to the commanders log screen. (Note, the load/save feature will be disabled the first time that this menu is viewed in a new game.)



Multi Player

Choosing the Multiplayer option from the main menu will present the player with two options, local multiplayer game and Internet multiplayer game.

Multiplayer Flow Chart



Local

Selecting a local Internet game will present the player with the option joining an existing game or hosting a new game. If the player hosts the game he controls the game rule options, if he joins he may only follow the rules set by the host. In either case the player will then be presented with the lobby menu screen.



Internet

Selecting an Internet game brings us into a slightly nebulous area. Some Internet services require that you terminate the application, connect to the internet, and launch an independent client application. Other services just require that you launch Internet explorer. Most likely we will provide connection options in the start menu that will allow the player to directly connect to these services. However if the player selects this from the list, we will close the Mech4 app and launch the default Internet connection service (most likely Mplayer.)

Lobby Screen Example

A schematic diagram of a lobby screen interface. It consists of several rectangular boxes representing UI elements. At the top is a single wide box. Below it is a large central box. To the right of the central box is a vertical column of four smaller boxes. At the bottom left is a wide box, and to its right is a vertical column of four smaller boxes. All boxes are empty, representing placeholders for content.

The lobby screen is similar in nature with that of MechCommander with a few modifications that are Warrior specific. Players will be able to adjust various game type options, control the number of players, allow for custom skins, set maximum 'Mech weights, etc. The lobby is designed to be similar to those used by online services.

It is important to note that each online service has their own matching lobby screen, as we offer support for online systems they player will see a slightly different lobby interface.

Also, as the amount of development time required for developing a score and ranking system is short due, the ranking feature will only be available when connected to an online service that will warehouse this data for us, such as Mplayer.



Options

The options menu will allow the player to change various specific rules and setting used in the game.

Difficulty

The difficulty can be set to easy medium and hard. These three levels will not change the players experiences in the simulation, (all the same missions will remain) but will adjust the level of AI in the enemies and encountered, and also the amount of damage targets require to be destroyed. In easier modes the targeting skills of enemies will be reduced as well as the amount of damage needed to cause critical hits. Building, turrets, and other objectives will also have a reduced (% based) amount of hitpoints.

Detail Complexity

This will be a very simple adjustment mechanism with low, medium and high modes. The game during the setup phase will suggest appropriate setting and will also be capable of adjusting detail levels of the polygon counts in real time (specifically the 'Mechs). This adjustment will reduce the size of the textures used, adjust the clipping plane for 3D scenes and limit the hi-level LOD's for lower class machine / video card situations.

Heat Rules

For the very novice player the option of turning heat rules off will make the simulation a much easier task to complete. In the off mode the player can run and shoot without any heat related consequences.

Introduction

Selection this option will replay the introduction animation for players who have already seen it but would like to view it again. Most likely this feature will be used by competing development companies as an example of "what I want my FMV to look like."

Credits

Place the entire development credit list for the project.

Quit

Quits the simulation and returns to the operating system.

Main Game Menu

The main game menu, (different from the main menu) is the primary menu screen used by the player during single player games. It will lead the player to game options, loading, saving, and mission areas.

Load / Save

Much like the initial load saved game screen from the main menu, selecting this option will allow the player to load a previously saved game. However, this version of the menu box will also allow



FASA INTERACTIVE

the player to save the status of the last completed mission. Note again, that a player will not be able to save a game mid-mission he MUST complete mission to be able to save.

Options

This is just another way of going to the main options screen and will function exactly the same as previously described above.

Commanders Log

The Commanders Log is the link into the mission simulation and technical details of the game. From the screen the player may review past missions, customize 'Mech in the MechLab, organize a lance roster, and launch a new mission.

In Game Menu

Pressing escape during gameplay in the simulation will bring up a very small option set that will allow the player to adjust the detail levels, or quit the current simulation.

Game Controls / Options

Interface Design – Keyboard

Below is the default layout for keyboard based commands. It is important to note that *all* of these controls will be customizable by the end user.

Weapons

(Chain-fire mode)

Fire Current Weapon	Spacebar
Select Next Weapon	,
Select Prev. Weapon	.

(Group Mode)

Fire Group 1	F1
Fire Group 2	F2
Fire Group 3	F3
Fire Group 4	F4
Fire Current Group	Spacebar
Select Next Group	,
Select Prev. Group	.

Toggle Group / Chain Fire	\
Add Weapon To Group 1	Shift+1
Add Weapon To Group 2	Shift+2
Add Weapon To Group 3	Shift+3
Add Weapon To Group 4	Shift+4
Jettison Ammunition	K

Targeting

Cycle Through All Targets	T
Previous Target	R
Nearest Enemy	E



FASA INTERACTIVE

© Copyright 1998 FASA Interactive Technologies, Inc.
MechWarrior is a trademark of FASA Corporation.
Confidential – For Internal Use Only – DO NOT DISTRIBUTE

Nearest Friendly	F
Under Reticule	Q
Inspect Target	I
Target Subsystem	Shift+T
Disable Targeting	Ctrl+T

HUD

Cycle MFD1 Mode	F5
Cycle MFD2 Mode	F6
Toggle On/Off Com Window	F7
Toggle On/Off Mission Parameters	F8
Toggle On/Off Nav Window	F9
Toggle On/Off Lance Orders Window	F10
Toggle On/Off Lance Status Window	F11
Save MFD / HUD Layout	F12
Toggle On/Off Cockpit	U
Toggle Sensor Mode	M

Mech Piloting

Throttle Control	1-0 (10% Inc)
Increase / Decrease Throttle	+/-
Steer Left / Right	⇐ ⇒
Reverse Direction	Backspace
Torso Twist	< >
Center Torso	/
Jump Jet	J
Select Nav Point	N
Autopilot	A
Toggle MASC	V
Shutdown	S
Override Auto-Shutdown	O
Eject	Ctrl+Alt+e
Toggle Auto-Ejection	Ctrl+e
Self-Destruct	Ctrl+Alt+x

View Controls

Zoom In	Z
Zoom Out	Shift+Z
Reset Zoom	Ctrl+Z
Look Left	keypad 4
Look Right	keypad 6
Look Back	keypad 2
Look Down	keypad 5
Look Up	keypad 8
Pilot Eye Control	Ctrl+⇐ ⇒ ↑ ↓
Thermal Optics View	L
External Camera	C
External Camera Controls	Ctrl+⇐ ⇒ ↑ ↓

Communications

Call up “Give Orders Window”	G
Activate Chat In Multiplayer	Tab



FASA INTERACTIVE

© Copyright 1998 FASA Interactive Technologies, Inc.
MechWarrior is a trademark of FASA Corporation.
Confidential – For Internal Use Only – DO NOT DISTRIBUTE

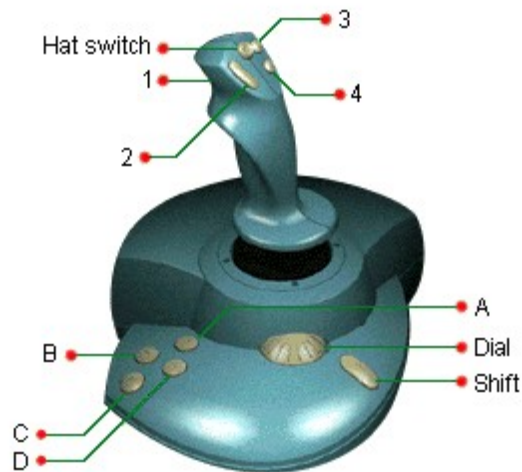
Other

Options

Esc

Joystick

The default supported joystick for Mechwarrior 4 will be the Microsoft family of sidewinder sticks. The joystick will typically work in conjunction with keyboard commands.



Button 1
Button 2
Button 3
Button 4
Button A
Button B
Button C
Button D
Hat Switch
Dial
Twist

Fire Current Weapon / Group
Cycle through all Targets
Select Next Weapon
Select Previous Weapon
Fire Group 1
Fire Group 2
Fire Group 3
Fire Group 4
Look Down, Back, Left, Right
Increase / Decrease Speed
Rotate Torso



‘Mech Sub-Systems

HUD

The HUD system will come in four basic types. The following information details the major systems that are available in the advanced sensor types.

The HUD can operate in two levels of transparency. The default “hi-end” mode will be a completely transparent image that also has translucent effects over the main view window. The slower machine, HUD mode, will only have transparency edges with each window cutout against a black background.

The primary HUD colors may also be customized based on the current mission location to maximize its view-ability.

Communications Video Window

Inbound communications from friendly or enemy targets are displayed via a small FMV playback. This video should be very basic, thumbnail sized, grainy images that are generic in nature and can be used with multiple audio files.

Mission Parameters Windows

This is a simple text window that describes the hi-level objective for the current mission and the status of each objective. (E.G. 1. Destroy enemy HQ – In Progress)

Nav Window

This is a small window that indicates what the current selected navigation point is, and the distance the player is from that nav point. (Eg. Nav Delta : 842 M)

Lance Orders

Up to four short lines of text can be displayed here that shows what each order the players lancemates are following.

#1	Guard
#2	Goto Nav 2
#3	Attack Target
#4	None

Lance Status

The lance status window will also show a simple graphic representation of the lances formation, and a hi-level summary of the each lancemates vehicle status in percentage, the color of the text will also be indicative of the status of each member to allow for fast player recognition.

Formation:	Chevron
Triptic:	80% (Green)
Davion:	35% (Red)



Nason: 50% (Yellow)

Target Data MFD Indicator

This window just indicates the mode the MFD is currently operating under.

Target MFD

The target MFD displays semi-detailed information regarding the object the player has targeted, if any.

This window will operating in a few unique modes. The availability of these modes is determined by the sensor package in the 'Mech. (See Sensors)

Tactical

In tactical mode the target will appear as a wire-frame over polygon mesh target viewing window. The mesh color will indicate what the general status of each sub section of the target it.

The player will be given the range to the target and the type of target in a simple text display, if the player is using an advanced sensor package, he will also receive the name of the target, the range to the target, the pilots name of target (if any), the rate of closure and what if any sub system is targeted.

Visual

The visual mode works just like an external camera on the target, showing the target in with a live camera feed.

The same text information will be available as in tactical mode.

IR

The IR targeting mode is an expansion of the tactical mode. Additional color information regarding the heat of the target will be represented by a heat aurora coming from the central area of the target. The entire image should be presented in what has become the standard IR shades of red.

The same text information will be available as in tactical mode.

Mech Status Data and MFD Indicator

This window just indicates the mode the MFD is currently operating under.

Mech Status MFD

This window breaks down the status of the player 'Mech. The basic display will be just like the tactical view used in the targeting MFD. The players 'Mech will appear in its current position with a wire-frame over poly model. The color of the wire-frame will indicate the general status of each subsection.



FASA INTERACTIVE

The player may then choose specific detailed information about 'Mech subsystems including Ammo remaining, damage, and armor status.

Weapons Area

The weapon area details the current status of each weapon the 'Mech is carrying. The color of the weapon name and box surrounding the weapon indicate what the status of the weapon is, (green – ready, yellow – charging, red – destroyed or out of ammo). If the weapon is an ammo based weapon the remaining ammo will also be displayed next to the name of the weapon. In all cases a small re-charge bar will indicate the recharging or charged status of the weapon.

HUD Rings (See Image for further info)

This is the center point of the HUD system in both literal terms and informational terms. The Ring will provide the player with a host of information designed to be easy to use for the beginning player, and providing extensive information for the advanced player.

The ring is made up of three independent rings.

Outside Outer Ring

The outer-ring contains information about the nearest enemy, nearest waypoint, the orientation of the torso, and all other objects detected by the 'Mech sensors.

The nearest enemy is “pointed at” by a red triangle that will indicate the direction of the enemy relative to the player.

Outer ring

The nearest waypoint is indicated by a gray triangle and pointed at in the same fashion.

Friendly targets are indicated by simple pixel pips in white

Other Enemy targets (not the closest) are indicated by a red pip.

Unknown objects are indicated by yellow pips.

Inside Outer Ring

On the left side of the ring is a digital readout of the 'Mech speed.

On the right side of the ring is a digital readout of the 'Mech temperature. Two pips will also indicate which mode the sensors are operating in (active / passive.)

On the upper side of the ring is a digital readout of the 'Mech heading.

On the lower side of the ring is a digital readout of the current targets range and rate of closure.

Middle Ring



Sensors

The sensor system provides players with long-range view of the world surrounding the player, but out of view of the naked eye. Effective use of sensor systems can make or break a 'Mechwarrior and if a sensor system is damaged, the player is at a huge disadvantage as they must then "fly blind."

Sensors come in two basic varieties, basic and advanced. Clan versions tend to have slightly improved range and effectiveness over IS.

Basic Sensors

Basic sensors have are run of the mill get the job done equipment. They have a limited technology base and will not support most advanced sensor features.

Active Mode (Basic Sensor Types)

This is the default mode for all sensor systems. In active mode, sensors can typically identify powered-up enemies up to 1500 meters. Active mode sensors can locate powered down Vehicles / 'Mechs at a range of 200 meters.

When a sensor is "switched" to active mode, it will power-up increasing its range from 500 meters extending to 1500 meters over a period of about 3 seconds. It is important to note that a passive sensor system will be able to detect an active sensor during the power-up phase which may be before the active sensor has reached its full range.

Passive Mode (Basic Sensor Types)

Sensors running in passive mode have limited range and are limited to line of sight sensing. However, they also help hide a player from enemies. In passive mode, sensors will typically locate enemies that are *running active sensors*, up to 1500 meters. Passive mode sensors cannot locate 'Mechs that are powered down.

Object Scan Mode

Active sensors can scan targeted objects that are in close range (less than 300 meters). A scan will provide detailed information regarding a specific object, such as the contents of the object or the name of an enemy 'Mech pilot, etc.

Targeting Modes – Basic Sensors

Tactical

Only the tactical targeting mode is available with basic sensor systems. This type of targeting is a wire-frame over polygon mesh target viewing window. The mesh will indicate the general status of each sub section of the target in color.

The player will be given the range to the target and the type of target in a simple text display.



Advanced Sensors

Active Mode (Advanced Sensor Types)

Active mode in advanced sensor systems works exactly like basic sensor types but with a slight longer range and with a slightly faster target locking cycle.

This is the default mode for all sensor systems. In active mode, sensors can typically identify powered-up enemies up to 2000 meters. Active mode sensors can locate powered down Vehicles / 'Mechs at a range of 250 meters.

When a sensor is "switched" to active mode, it will power-up increasing its range from 500 meters extending to 2000 meters over a period of about 2.5 seconds. It is important to note that a passive sensor system will be able to detect an active sensor during the power-up phase which may be before the active sensor has reached its full range.

Passive Mode (Advanced Sensor Types)

Passive mode in advanced sensor systems works exactly like basic sensor types but with a slight longer range, advanced types and also have a faster target locking cycle.

Sensors running in passive mode have limited range and are limited to line of sight sensing. However, they also help hide a player from enemies. In passive mode, sensors will typically locate enemies that are *running active sensors*, up to 2000 meters. Passive mode sensors are based on heat and will detect other 'Mech's / Vehicles that are within 1200 meters. Passive mode sensors cannot locate 'Mechs that are powered down.

One major benefit to passive mode sensors is that they can be run while a 'Mech is in a powered down state, perfect for leading (or being lead into) a trap.

IR Mode – (Advanced Sensor Types)

Advanced Sensors in either active or passive mode can be switched to an infrared mode, which will change the main view window to a basic red hue, useful mostly in night missions. All objects in the world will be displayed in shades of red, based on heat levels. The higher the heat of an object the brighter and lighter it will appear. Note: these colors will be scripted within the game engine, not simulated.

Targeting Modes – Advanced Sensors

Object Scan Mode – Advanced Sensors

The object scan mode work exactly like Active sensors can scan targeted objects that are in close range (less than 300 meters). A scan will provide detailed information regarding a specific object, such as the contents of the object or the name of an enemy 'Mech pilot, etc.



Tactical Targeting (Advanced Sensor Types)

The tactical targeting mode for advanced sensor systems works exactly like the basic sensors system targeting mode, but will provide a bit more detailed information to the player.

The player will be given the range to the target, the type of target, the name of the pilot, in a simple text display.

IR Targeting Mode (Advanced Sensor Types)

The IR targeting mode works essentially the same way as a full viewing mode, however the target window will show the overall heat signature of the enemy target. The object will appear just as a tactical view, but a hue of heat signature will expand and change color emanating from the center of the object. The color spectrum should range from blue (cool) to red (hot), to white (about to explode.)

Visual Targeting Mode (Advanced Sensor Types)

Just as it sounds, a visual targeting mode will put a “camera” like viewer on the object and its immediate surroundings. The same text information will be provided to the player, but no colored wire-frame information will be available.

Subsystem Mode

This is the major advantage of the advanced sensor system. Once an object has been targeted, the player may choose to target a subsystem of that target. On a ‘Mech the major subtargets include, each arm, each leg, torso and head.

Advanced Sensors / Electronics

As more advanced ‘Mechs become available to player, so will more advanced electronics systems. These systems aid in the detection of enemies, the directing of weapons and counter weapons systems.

Beagle Active Probe (BAP Requires Advanced Sensors)

A BAP allows a basic ‘Mech sensor system to see farther and reveal what may be hidden. The BAP will detect all ‘Mechs, in both passive and active sensor modes and will also detect powered down ‘Mech’s.

The BAP is defeated by an ECM suite and will only work in active sensor modes.

C3 Computer (Requires Advanced Sensors)

The C3 Computer system is essentially a simple wireless network that a lance of ‘Mechs use to share sensor data between the lance members. If one ‘Mech has an enemy within scanning range, in a lance that has the C3 system, that data is shared with all other members of the lance. If one can see it, all can see it. The system contains two elements - a master computer and a series of slave computers. If the master is damaged or destroyed, the entire system fails, whereas if a slave computer is disabled, it has no effect on the others in the lance. When a C3 master unit is destroyed, all ‘Mechs in the lance will simply operate with the default independent sensor systems.



Guardian ECM (Requires Advanced Sensors)

The Guardian Electronic Counter Measures system (ECM) works against basic and advanced weapons and sensor systems. The Guardian system will nullify the effectiveness of the C3 computer system, the Beagle Active Probe, the NARC missile and the Artimis FCS. Further, when a ‘Mech comes within 100 meters of another ‘Mech with the Guardian ECM system, the ‘Mech sensors will only produce static in the radar and targeting windows.

A ‘Mech with ECM is also more difficult to lock onto and will add one (1) second to the normal amount of time it takes for a targeting computer to lock on.

Artimis IV FCS (Requires Advanced Sensors)

The Artimis IV Fire Control System allows players to target an enemy faster than the normal 3 seconds it takes for lock on. With the Artimis IV system a player can lock onto a targeted objects in just 1 second. (Note: the Guardian ECM system negates this effect.)

NARC Beacon (Requires Advanced Sensors)

The NARC beacon is a player guided (fly by wire) missile that attaches itself to any metallic object (‘Mechs and Vehicles). When a NARC is activated by the player, the main HUD view will change to the view of the NARC missile. The player can then steer the missile with normal interface controls (mouse or joystick). If the NARC is flown into an enemy object, that object is highlighted on every ‘Mech sensor system. A NARC’d enemy can be targeted and locked onto from a range of up to 500 meters regardless if it is in the line of sight or not. The NARC also provides good medium-range visual reconnaissance.

Tracking Systems

Tracking is handled automatically by the simulation. When a ‘Mech, special object or vehicle comes into range of the sensors, the object is automatically tracked. The simulation will be able to track as many objects (probably never more than 32) that are within the range of the sensors.

When a object is detected by the sensors, a dot will be placed on the sensor display of the HUD in relative location to the player. The dot will indicate what type of object is being tracked. A blue is a neutral or unknown object type, green is a friendly object or ‘Mech, Red is an enemy object or ‘Mech.

Targeting systems

The player can target any object that is within visual range and there will be a quick key and/or button commands to allow for rapid targeting of nearest friendly and enemy objects. Only one object may be targeted at a time.

Depending upon sensor mode, the player will receive additional information about the target as follows:

<u>Info</u>	<u>Description</u>	<u>Available</u>
Pilot Name	“Paladin”	“Scanned” in Active Mode



FASA INTERACTIVE

© Copyright 1998 FASA Interactive Technologies, Inc.
MechWarrior is a trademark of FASA Corporation.
Confidential – For Internal Use Only – DO NOT DISTRIBUTE

Target Type	“Hunchback”	Active Mode
Target Range	“859m”	Active / Passive Modes
Visual Appearance	(see HUD)	Active Modes
Team/Lance Name	“Dave’s Warriors”	Active Mode (multiplayer)

To lock a “steering” or “in-flight tracking” weapon, such as a missile, the player must move the targeting reticule over the object and hold it there for 3 seconds while the target remains in range and is currently the targeted object. The amount of time it takes to lock on can change based on the type of sensor system the player ‘Mech has and if there are any ECM’s in the area.

A player can lock onto a target in front of the ‘Mech via the normal front view or they can lock onto enemies to the right or left of the player ‘Mech by looking left and right. When looking to the sides of the player ‘Mech, only the respective arm will be come into view. Externally both arms will track to the direction the pilot is looking.

If a ‘Mech has an advanced sensor system, a targeted object will have a leading icon in addition to a targeting bracket (see HUD). This leading icon is the exact direction the player should fire slower moving projectiles to achieve a “hit” on the targeted object so long as the object doesn’t change speed or direction.

Virtual Cockpit

In an effort to bring realism to the simulation and also to make the player “feel” like they are inside of a moving vehicle, we will create unique “inside” cockpit views for each ‘Mech.

The default view will be oriented from the pilot’s seat looking forward through the front windshield. While each cockpit will be different, these views will only show the inside of the support pillars as the pilot would see them. We will not be displaying dashboard gauges. All critical information can be found within the HUD.

These inside views will also provide the player with tactile feedback regarding his situation. The bobbing motion cockpit will indicate how fast the player is traveling. When the player takes damage, the cockpit will rock, away from the impact, indicating which direction the player is taking fire from.

A great benefit gained by using a virtual cockpit is the fact that we can use this view for cool visual effects. The pilot’s reflection will appear faintly on the glass and as the ‘Mech takes damage, lights will appear to reflect on both the glass and pillars indicating that warning lights are flashing within the cockpit.

The virtual cockpit theme will also encompass the area to the left and the right of the pilot. If the player chooses to look left or right the cockpit will quickly scroll or snap to a view looking out through the sides of the cockpit. The HUD will remain active and the player will be allowed to fire upon objects to the sides of the ‘Mech, but only with weapons that are attached to arms on the respective side of the ‘Mech.

If a ‘Mech has shoulder mounted weapons, which may obscure the view from the sides of the cockpit, the simulation will cheat the weapon mounting back so that the player can still use this feature for consistent gameplay and combat reasons. The player will still be able to identify that the weapon is present, it will just block the view.



FASA INTERACTIVE

Movement and Animation

The animation system in MW4 will be environment smart. A ‘Mech will intuitively know how to walk over difficult terrain and how to respond to various terrain changes. For example, when a ‘Mech walks over a ledge, it will not continue to “walk” but will move its legs into a landing position. ‘Mechs will also be able to walk over and on special terrain objects. Small rocks, destroyed buildings, trees and other cultural objects can be traversed by the player’s ‘Mech.

Walking / Running

The BattleMech has the ability to travel over difficult terrain. As such, the animation system will allow player ‘Mechs to travel up and down fairly steep terrain. ‘Mechs will be able to climb slopes of about 45 degrees and descend slopes of any degree. When going to very steep downhill areas, the motion essentially becomes a downhill leap or long jump.

To accomplish this task visually we will employ a smart animation blending technology that will allow the ‘Mech to move smartly over a variety of terrains.

- Walk (up / down / flat)
- Walk Backward (up / down / flat)
- Run / Stride (up / down / flat)



Torso

Most ‘Mechs can also move the torso and arms independent of the legs. While running or walking, the ‘Mech can rotate its torso and attune it to a target. Each individual ‘Mech will have a unique torso twist angle and twisting rate.

When firing or taking damage, the ‘Mech torso will recoil in the appropriate direction and range to the cause.

- Tilt torso
- Rotate torso
- Rotate arms
- Tilt arms
- Recoil (procedural, used for both damage and weapons fire)

Arms

Arms will have two primary joints at the shoulder and wrist locations, which will allow the arms to track targeted objects independently to the torso, when firing the arms will also recoil slightly to emphasize the power of the ‘Mech weapons.

- Tilt arm
- Rotate arm
- Tilt hand
- Rotate hand
- Recoil (procedural)

Special Movements

‘Mechs will also have a few specific movements mostly used for visual effect.

As a ‘Mech takes damage, particles and actual pieces of armor and perhaps limb pieces will animate as they fly off the ‘Mech.

When taking significant impact damage, the ‘Mech can actually be knocked over in eight directions, although the ‘Mech will always land on in front. Once a ‘Mech has been knocked down, it will probably be rather damaged but can still get up as one as at least on leg is in working condition.

‘Mech legs can be damaged in two different ways. When a leg takes a critical hit, the leg will can be “Gimped”, forcing the ‘Mech to drag the damaged leg while walking. If both legs of a ‘Mech are gimped the ‘Mech will be considered killed.

‘Mech ankles will allow the player to rotate around one foot when standing still, avoiding the lumbering turn-around found in the ‘Mech 2 game series. This rotation at the ankle will also be used when the ‘Mech is walking very slowly. (See MW3 zipper demo for reference)

When jump jetting from standstill, the ‘Mech will crouch and jump up just as the jump jets fire. While in flight, the ‘Mech will remain mostly still in a standing like position with one leg slightly elevated.



As ‘Mechs can move at very high rates of speed a hi-speed animation will be created that basically represents a ‘Mech doing an impression of a snow plow.

Significant damage to a standing ‘Mech will cause a cockpit ejection animation. When a ‘Mech has been ‘killed’, the POV of the camera will quickly move outside and above the player ‘Mech and the ejection pod will shoot off from the top of the ‘Mech. One should not attempt this under large objects or while on the ground.

Last, but certainly not least, each ‘Mech will have a massive death explosion animation where numerous geometry and particle pieces rain from the explosion. We’ll trust our animators to have fun with this.

- Stand (pose)
- Fall (forward or rotate to forward during fall)
- Crouch (pose)
- Stand up
- Gimp walk
- Rotate on foot
- Jump jet
- Land
- Eject
- Catatonic death
- Hi-Speed Fall

A number transitional animations will be created to provide smooth motion across all animations.

- Stand to walk
- Stand to walk back
- Stand to run
- Run to stand
- Walk to stand
- Walk back to stand
- Walk to gimp
- Stand to gimp

Heat Generation

Heat is the enemy of every ‘Mechwarrior. Effective heat management is critical to the success of almost every mission in the game. If a player doesn’t pay close attention to the status of their ‘Mech’s heat, they will soon find themselves quite dead.

‘MechWarrior IV will track heat similar to MW2 and other previous versions of ‘MechWarrior games. Heat will be expressed in degrees Kelvin, which will be based on the heat levels found in the BattleTech board game and then modified as deemed needed by gameplay.

Environment

Ambient temperatures in game environments will effect the ‘Mech’s heat level. A hot desert will increase a ‘Mech heat level, as will standing in a lava pool or in the burning



rubble of a destroyed building. Conversely, cooler planets and water will have a positive effect on heat management.

Weapons

The use of any weapon will create heat in a 'Mech. Naturally, energy-based weapons such as the PPC or lasers will typically generate a great deal of heat. Projectile weapons such as machine guns, auto-cannons and even missiles also generate heat.

Engines (movement)

The faster a player moves, the more heat is generated by the movement systems. Running alone will not generate a serious level of heat, but running *and* shooting will.

Damage

When any type of weapon strikes a player, the heat level will increase. Depending on the weapon type, this effect can be rather minimal (machine gun) to severe (PPC's, lasers and missiles).

Heat Sinks

A 'Mech radiator. The more heat sinks on a 'Mech, the better capable the 'Mech is of handling heat. A single heat sink will reduce a 'Mech's heat level by a certain number of degrees per second. A double heat, as its name indicates, will reduce the heat twice as fast as a normal heat sink. During pre-production, the specific rates will be determined.

Coolant flush

The player can flush excess coolant through the heat sinks in order to rapidly reduce the heat level of a 'Mech. This is typically used in the middle of a battle when the player's 'Mech is hovering close to critical temperature. Flushing coolant through the heat sinks will double the heat sink's effectiveness for as long as the player continues flushing coolant. The amount of coolant will be limited to a total of 4 seconds of flush.

Heat Effects

As the heat level of a 'Mech increases, major 'Mech systems are adversely effected. The higher the heat level, the greater the problem for the player.

Shutdown

When a 'Mech reaches critical heat levels, it will automatically initiate a shutdown procedure in order to avoid ammo explosions or worse reactor meltdown. However, this action tends to happen at critical times in the simulation and can therefore be overridden by the player.

Ammo Explosions

Once a 'Mech has gone beyond the critical temperature threshold, it runs a random chance of detonating on board ammunition as a result of the high heat situation. Ammo



FASA INTERACTIVE

© Copyright 1998 FASA Interactive Technologies, Inc.
MechWarrior is a trademark of FASA Corporation.
Confidential – For Internal Use Only – DO NOT DISTRIBUTE

explosions can cause significant damage to both the pilot and the 'Mech, depending on the quantity of ammo detonated.

Meltdown

The pinnacle of bad judgement. If a driver allows his heat level to remain beyond the critical threshold for an extended period of time, the BattleMech will, without doubt, explode in a very extravagant fashion completely destroying itself, the driver and just about anything within 20 meters of the 'Mech.

Reduced Speed

As a 'Mech's heat increases, its ability to run at maximum speed will be gradually reduced up to 20% as the 'Mech temp reaches into the critical heat levels.

Sensors

Sensor range will be reduced by 10% when the level is halfway to critical. At critical levels, sensor range will be further reduced by an additional 10%.

Weapons

'MechWarrior IV will support most of the weapons found in the paper version of the BattleTech game, but in the interest of keeping the game fast and fun, certain elements will be simplified. Inner Sphere weapons will typically be the basic versions, while Clan weapons will typically be extended range or special version weapon types.

(Full details of the weapon's size, weight, damage, range and recycle will be finalized during pre-production.)



Energy Weapons

Lasers

Pulse Lasers

A pulse laser is a fast moving, glowing projectile. The pulse laser fires in 3 round bursts. Pulse lasers will appear as glowing light beams with smoky haze animations around them. Each version has a unique color (small, medium, large).

- Small, Medium, Large

Beam Lasers

A beam laser is an instantaneous weapon that creates a beam of highly energized light particles. The beam will cause damage based on the amount of time it strikes an object. A laser will discharge over a period of about 3 seconds.

(For networking reasons, we may not allow this weapon to act instantaneous, but rather just give it a high rate of velocity).

The beam will appear as an extended version of the pulse laser.

- Small, Medium, Large – Basic
- Small, Medium, Large – Extended Range

PPC

The particle projection cannon consists of a magnetic accelerator firing high-energy ion bolts that have a devastating effect on most targets, specifically 'Mechs. The PPC is the most effective weapon to use against 'Mechs, but produces considerable heat, is heavy and large.

The PPC will appear as a writhing, living energy pulse-like beam about 10 meters long that will travel fairly slow across the terrain. When an object is hit by a PPC bolt, it will cause a very large, blinding flash that will rapidly fade and show the energy of the bolt dissipate off the object it stuck (assuming the object survived the hit).

- Basic, Extended Range (Clan)

Projectile Weapons

Autocannons

Basic Autocannons

Autocannons fire high-speed streams of high explosive, armor-piercing shells. When fired, these will appear as a short burst of bright tracer fire in the game. Based on the particular configuration, an autocannon can do up to 20 points of



damage (AC/20), based on the number of actual projectiles that collide with the targeted object.

- AC/2, AC/5, AC/10, AC/20

LBX Autocannons

The LB-X version of the autocannon is essentially a ‘Mech-sized shotgun. It fires a burst of smaller and lighter cluster munitions in a tight spread pattern. This increases its chances of hitting an enemy but will typically reduce the overall damage it causes.

- LB-X/2, LB-X/5, LB-X/10, LB-X/20

Ultra Autocannons

Functionally, Ultra Autocannons work exactly as normal Autocannons, but they can fire ammo at twice the rate. Within the game, these will appear to fire as normal, but with twice the amount of tracers (in density) and will do 2x the damage.

- Ultra AC/2, Ultra AC/5, Ultra AC/10, Ultra AC/20

Missiles

Numerous missile systems will be available in MW4. Each has a slightly different and unique appearance and effect within the game system. The missile itself will appear as a rather slow moving explosive projectile and can ‘lock’ onto a target.

Similar to the MW3 design, all missiles in MW4 will have the ability to lock onto targets. Further, rather than forcing a player to lock each individual missile type onto an enemy, once a lock has been acquired it will be locked for all missile and weapon types. It is important to note that missiles can be fired without a locked target. Sometimes in the heat of a battle this is a smart way to knock out an opponent.

(See Sensors for more information on target lock on.)

LRM

Long-range missiles appear to fly in a fairly high trajectory arc. Each missile will create an animating card or particle smoke trail that will slowly dissipate and also have a light point at the engine side of the projectile simulating a rocket engine burn.

Depending on the missile pack configuration, various salvo sizes will be launched. By firing numerous missiles in a salvo, the odds of scoring a hit with one or more of the missiles is increased.

Individual missiles don’t do much damage to a target, but are generally launched in a salvo that increases the odds of hitting a target because missiles will spread out slightly while in flight. A well-aimed shot could cause a large number of hits against a target with only one salvo launched, which also balances the relatively low impact per missile. ‘MechWarrior IV will also allow additional



damage to be inflicted by missile explosions that occur in close proximity to the target, but don't actually hit the target, known as splash damage.

- LRM/5, LRM/10, LRM/15, LRM/20

SLRM

The streak version of the long-range missile developed by the clans and operate under the same basic rules as the SRM version. In the game, SLRM's will be very rare and will take a fairly significant amount of time to lock on to an enemy target.

- SLRM/5, SLRM/10, SLRM/15, SLRM/20

MRM

MRM missiles are small non-guided medium range missiles. As they are "dumb" missiles what they lack in accuracy they make up for in sheer number.

- MRM/10, MRM/20, MRM/30, MRM/40

SRM

Short Range Missiles will appear exactly as their long-range counter parts, but will fly in a flat trajectory and at a higher rate of speed.

- SRM/2, SRM/4, SRM/6

Streak SRMS

The streak version of the short-range missile can only be fired when a target has been locked. The major benefit to the streak missile is accuracy. Once a lock has been acquired, the missile will not miss, so long as the target stays in the missile's line of sight. The only drawback to streak missiles is weight, which is 50% higher than normal SRMs.

Streak SRMS appear as normal SRM, but travel with a slightly higher trajectory arc and move much slower.

- SSRM/2 (IS)
- SSRM/2, SSRM/4, SSRM/6 (Clan)

Thunderbolt

Gauss Rifle

The gauss rifle uses a series of magnets to propel a very heavy metal projectile at very high speeds over great distances.

The power of this weapon is incredible. It can literally throw a 200 kilogram slug almost a kilometer and creates very little heat in the process. The only downside of this weapon



FASA INTERACTIVE

is it has a very long recharge time and its ammunition is very heavy, limiting the number of rounds any 'Mech can carry.

The projectile itself travels at a velocity that makes it invisible, although it will leave a smoke trail similar to the effect seen in the film Eraser.

Machine Guns

While not a weapon, a 'Mech typically carries the machine gun can be effective against ground troops, but does little or no damage to typical 'Mech armor.

The machine gun visual will consist mainly of muzzle flashes and ricochet effects such as sparks or ground puffs, depending on the target.

Other Weapons

Flamer

The flamer is a heat-based weapon that emits a short-range burst of extremely hot gas, which is super heated by venting the gas through the nuclear engine compartment. While it has a very short effective range, it is highly effective against ground personnel.

LAMS

The Laser Anti-Missile System is a solely defensive system used to defeat incoming enemy missiles. A 'Mech with a LAMS can destroy up to 25% of an incoming missile volley. Unfortunately, this system tends to produce a fairly large amount of heat and can hamper the offensive aspect of some 'Mech designs. Further, LAMS is defeated by a Guardian ECM system within 100 meters.

AMS

The basic AMS is a machine gun version of the LAMS system. The system has the same basic ratings as the LAMS but doesn't create nearly as much heat. The downside of course is the need for ammo.

Weapon Chart

See attachment

Damage / Death

A 'Mech is divided into a few specific sub-sections, each of which take damage slightly differently and may or may not cause the destruction of the 'Mech or perhaps death of the pilot. Every 'Mech will contain 2 Legs, 2 Arms, 3 Torsos (Left, Right, Center) and a Head section. For purposes of armor, each torso section will also contain a front and back section. These sections contain a wide variety of elements which can be destroyed independently, such as a specific weapons, and not cause the destruction of the sub-section or result in 'Mech death.

Armor



A armor also plays a critical role in how a 'Mech takes damage. Armor is handled generically by location and varies dramatically based on the 'Mech type. That is, we will not distinguish between inner structure armor and outer structure armor - only by overall armor values. If the entire armor pack on a particular sub-section is destroyed, that sub-section is also destroyed. If a leg or other limb has its entire armor completely destroyed, it will be destroyed. If the cockpit or the entire torso section is completely stripped of its armor, the 'Mech will die.

A armor is reduced by incurring weapons fire on a point - for - point basis. For instance, if a player takes a 5-point hit in the torso, 5 points of armor will be taken from the torso section. After the armor is removed, a critical hit may occur based on the percentage of armor remaining.

Every 'Mech will come with a standard armor set in a default allocation pattern. Players can then add or remove this standard armor, which will not effect critical spaces. However, adding armor will increase the weight of the 'Mech.

Ferro Fiborous Armor

Ferro Fiborous Armor will increase the overall effectiveness as a percentage factor of the entire armor system on the 'Mech. This type of armor is not allocated into specific areas. To use it, the player must only meet the overall critical space and weight restriction for the 'Mech. This armor type will remain on the 'Mech until the player chooses to remove it or the 'Mech that carries it is destroyed.

Critical Hits

As a 'Mech's armor is reduced, the odds of a critical hit to a specific system increase. In 'MechWarrior IV, each specific system (engines, legs, etc) can take a certain number of hits (1-3) before the system is destroyed. Following level two rules, specific items such as gyros, engines and sensors will remain only damaged until over 50% of the critical spaces occupied by the item are destroyed. All other equipment types will be completely destroyed when they receive a critical hit. When a critical space filled with ammunition is hit, all the ammunition remaining in the space is destroyed. If the space is not equipped with the CASE system, significant collateral damage may occur.

Critical hits occur randomly based on the amount of armor that a specific hit region has remaining. We may allow multiple, critical hits from a single hit depending on the amount of overall damage caused by the weapon.

Critical Hit %	
<u>A armor Remaining</u>	<u>Random Hit Factor</u>
100% - 51%	0%
50% - 31%	20%
30% - 10%	35%
1 - 9%	75%
0%	Destroyed

Critical Hit Locations



FASA INTERACTIVE

When a critical hit occurs in a particular location, it will randomly effect one of the systems contained within that location.

Head

The head of the 'Mech will always contains three primary systems and typically one free critical space.

Sensors

A critical hit to the sensor system will either damage or destroy the system. A sensors effective range will be reduced by 50% each time it suffers a critical hit and will produce false signals (obvious to the player) until it is destroyed. Once destroyed, only static will appear in the targeting window and sensor display. Navigation will remain operating as normal to keep the player from getting lost.

Cockpit

A single critical hit to the cockpit will destroy the cockpit. If the cockpit is destroyed, the 'Mech is considered dead, as is its pilot. Shoot for the head.

Life Support

It takes two critical hits to knock out the life support system. Once the system is destroyed, the pilot will fry by the heat generated from the nuclear power plant used in the vehicle.

Torso

The torso is broken into three sections - left, center and right. In typical configuration, the left and right sections mostly contain weapons and heat syncs, although some engine and gyro types (XL) must contain critical spaces in all three torso sections.

It is important to note that if any sub-section of the torso is destroyed, it will cause the destruction of the entire torso section. As such, it is important to know which side of the player's 'Mech should face the enemy.

Torso Center

Engine

The center section will always contain either the entire engine or at least the main portion of the engine. If more than 50% of the engine in this area is destroyed, the 'Mech will explode, killing the 'Mech and possibly killing it's pilot.

Gyros

The gyro system of each 'Mech is proportional to the size of it's engine. If more the 50% of the gyro criticals are damaged, the gyro is destroyed. When the gyros are destroyed, a 'Mech will not be capable of jumping and will appear to bob more from within the virtual cockpit view.



Other

There are typically a few remaining critical spaces in the center torso area which can contain just about any type of equipment.

Torso Right

This area is typically filled with specific equipment damage and destruction of which is handled on a crit by crit basis.

Torso Left

This area is typically filled with specific equipment damage and destruction of which is handled on a crit by crit basis.

Legs

The leg area has a few required elements and a number of available critical spaces. Legs contain a hip and three actuators. If any of these elements are destroyed, the leg will be gimped. However, if all the leg armor is destroyed, the leg will be destroyed and completely removed from the 'Mech.

Arms

The arm area also has a few required elements and a number of available critical spaces. It typically contains the shoulder piece, two arms and possibly one hand actuators. Much like the legs, if any of these critical elements are destroyed, the arm will be gimped or destroyed.

Transfer of Damage

If a 'Mech has only a stub remaining from a destroyed arm or leg, and the stub is hit by weapon fire, the damage incurred by the hit is transferred to the appropriate torso subsection.

Jump Jets

The very popular death from above move found in the previous version of 'MechWarrior will also be included in 'MechWarrior IV. Damage from heat will occur if a jump-capable 'Mech lands on top of another 'Mech.

Splash Damage

A 'Mech can also take damage from effects that occur in close proximity to it. If a physical weapon detonates near to the player's 'Mech, it can potentially take damage commiserate upon the distance from the detonation. The element of damage can also include damage from falling debris, which can range from small pieces of a damaged 'Mech, up to massive boulders or chunks from large buildings, which can potentially destroy a player's 'Mech.

Engine Output / Generator System



FASA INTERACTIVE

© Copyright 1998 FASA Interactive Technologies, Inc.
MechWarrior is a trademark of FASA Corporation.
Confidential – For Internal Use Only – DO NOT DISTRIBUTE

For experienced 'MechWarriors only. Each 'Mech can allocate its engine output to specific generator systems within the BattleMech. We will simulate three independent systems within the 'Mech that players can customize during gameplay to suit particular situations.

Initially, the outputs will be evenly split between the three systems. From this point, the player can modify the output by increasing or decreasing power to one system at a time. As the player does this, the power being sent to the other two systems will be evenly adjusted. (Much testing needed)

Myomers

Increasing output to the myomers will increase the top speed of the 'Mech (up to 120% of top speed). This is generally a good idea when fleeing from a powerful enemy or in situations where movement takes precedence over seeing and shooting.

Weapons

Increasing the output to weapons recharge will allow energy-based weapons to recharge up to 20% faster. This is very useful when the player chooses to quickly dispatch an enemy, but will cause severe heat problems if the battle drags out more than a minute.

Sensors

Increasing output to sensors will increase the effective range of active sensors up to 20%. This is most helpful for 'Mech's that are performing recon-based missions.

It is important to note that adding output to any specific system will effect the other systems. Increasing output to one system will adversely effect the other two systems. This feature should be used with great caution.



Resource Management

‘MechWarrior IV will include an element of resource management for the player to consider. Our intention is not to introduce the element purchasing into the simulation. Rather, the player will be given a finite amount of resources to work with and must manage the usage of those resources. Resource management will come in two forms in the simulation, dependant upon which campaign the player has chosen.

Steiner Loyalists

The Steiner loyalists have full support from the Royal house and therefore will have significant resources to draw upon. Resource management will have a reduced role in this campaign of the game. Before each mission, the player will receive a shipment of goods allocated to the player for the specific mission. Based upon the contents of the shipment, the player may choose which ‘Mechs they will bring and how they are outfitted. Naturally, this provides the designers the ability to provide the player with new technology, weapons or ‘Mechs as they get farther into the game.

Davion Loyalists

The Davion loyalists are much more limited in terms of resource availability. They will not have a vast amount of resources to work with and will only receive shipments every 3 –5 missions. As such, careful management of resource inventory will be critical to succeed in this campaign.

Persistent Elements

Because there will be no bartering or purchasing system in the game, we will track specific resource elements persistently therefore, increasing their value in the game. If the player loses a critical item, it's gone forever.

Very basic elements, such as small lasers, will be abundantly available in the game, but hi-tech and hi-value items, such as a clan ER PPC, will be rare and therefore, valuable during the simulation. Also, because of the no money system, the element of repair fees will be eliminated. As such, damage to individual items will be treated differently on a case by case system.

Here is how we will track elements in the simulation:

‘Mechs / Vehicles

If a ‘Mech is destroyed during the simulation, it will be lost. If a ‘Mech returns from the simulation with damage, such as a missing or damaged limb, it will automatically be repaired at no cost to the player. ‘Mechs are extremely valuable items and losing a big one in the simulation will be a very big deal to the player.

Pilots

A pilot that is killed during the simulation is lost forever. This can pose as a serious threat to the player if the lost pilot was high in overall skill levels.

Engines

Damaged engines will be automatically repaired at the end of a mission. An engine will be considered destroyed if half or more of its critical spaces have been destroyed.



FASA INTERACTIVE

© Copyright 1998 FASA Interactive Technologies, Inc.
MechWarrior is a trademark of FASA Corporation.
Confidential – For Internal Use Only – DO NOT DISTRIBUTE

Regardless, if an engine is destroyed it will most likely result in the explosion of the 'Mech, leaving the player with very little to consider repairing anyway. Every 'Mech will come with an engine. Only the hi-tech engines, such as XL versions, become valuable to the player.

Sensors / Equipment

Treated just as engines. If it's damaged, it will be automatically repaired. If it's destroyed it will be lost. Basic sensors will be abundant. Advanced systems rare. All other equipment such as CASE and MASC systems will be handled on a case by case basis.

Armor

Armor will be treated uniquely. The player will have a large quantity of basic armor in the game, advanced armor such as the ferro-fibrous type will be rare. Regardless of the type, armor will be automatically repaired in the game and will only be lost to the player if the 'Mech carrying it is lost.

Ammo

Ammo will be tracked on a per round basis. When a 'Mech returns from a mission, the remaining rounds will be added to the resource pool.

Salvage

Both campaign modes will support a complete salvage system. Captured equipment and 'Mechs will be added to the player's resource inventory. Vehicles will always have a random element of destruction when killed. If a vehicle is destroyed, all of its contents will be lost. If a vehicle is killed but not destroyed, it will become salvage and a random amount of its equipment, weapons and remaining ammo will be added to the player's inventory as salvage.

Friendly pilots can also be rescued in the simulation and will be added to the player's pilot pool, so long as they are not killed during the rescue attempt.



Networking

‘MechWarrior IV will contain basic networking elements designed to allow for quick and easy deathmatch based multiplayer games that can include at least 8 individual players. Based on testing, we may increase the total number of players to 16, which will allow larger scale matches in each game type. These games will be supported in both local and Internet modes.

GAME TYPES

Deathmatch

A free for all deathmatch is just that, everyone for themselves. A deathmatch game will only end when either a maximum time limit is reached or a maximum kill count has been reached. In this mode, when a player is killed, they will automatically be returned to the simulation at a predetermined start point.

Team Deathmatch

Just like a normal deathmatch, but teams can compete against each other. Teams can consist of 2 sets of 4 players or 4 sets of 2 players (based on 8 max. players). The game will still only end when based on time and kill counts.

Arena Match

An arena game is similar to a deathmatch, but only allows for one death per player. In this game mode, if the player dies, it's over. The player may still view the battle as a spectator, but they will not be able to participate in the battle until a new match starts.

Team Arena Match

Just like a normal arena game, but allows for team vs. team play. In this mode, when all the players from one team are killed, the match ends.

Capture The Flag

Capture The Flag is a derivative of team vs. team multiplayer. In this mode each team searches for the opposing teams base which contains it's flag. To score in this mode, Team A must locate and take the Team B flag back to the Team A base *while* the Team A flag remains at the Team A base.

If a player carrying an opposing teams flag is killed, the flag is dropped. Once the flag has been dropped, it will remain there until the opposing team can pick up the flag and continue to run it to their base or the team who owns the flag picks it up, at which time it will return to the home base.

LOBBY SYSTEM

‘MechWarrior 4 will use a lobby system to simplify the process of meeting other players and setting up specific game parameters. The lobby type will differ slightly when playing multiplayer games locally vs. over the Internet.



FASA INTERACTIVE

© Copyright 1998 FASA Interactive Technologies, Inc.
MechWarrior is a trademark of FASA Corporation.
Confidential – For Internal Use Only – DO NOT DISTRIBUTE

The player who initiates the lobby, the lobby host, has exclusive control over the variables of the game. The variables include: if the game is a public or private password protected game, the maximum number of players, the type of game that will be played, the level that the game will be played on, any specific rules for the game, the maximum 'Mech weight and if the game lobby will remain open after the combat has begun.

Before the game has begun, the lobby will list the players that are active in the lobby and will allow for chat between all the players in the lobby via both voice and text chat systems. A player can chat to all the members of the lobby or 'whisper' to one specific player in the game.

Individual players can also set up their unique settings while in the lobby. Individual settings will include selected 'Mech types, camouflage settings, logo settings and team affiliation (if any).

Once the game settings are finalized, the lobby host can start the actual combat. When this happens, all the players in the lobby will be jump-ship dropped onto the game world and the match will begin.

In the event that the host leaves or quits the game after the combat has begun, the host, or server, will be migrated to the next active player in the network or the player with the lowest overall ping time. The game will not end when the initial host leaves, it will only end based on the game type or if less than two players are present. When a match does end, all the players who were active in the match are returned to the game lobby.

Local

In a local area network, the lobby operates independently from any larger hosting system (such as Mpath). Local networking games can be hosted or joined by anyone with a valid copy of 'MechWarrior (DVD in drive) within an individual network domain.

Internet

The Internet lobby system is slightly more complicated, as it is usually a sub-section of a larger online system. To connect to an online multiplayer game, the player must connect to a game master server typically located on an online gaming service.

We are considering developing our own lobby server or FASA Master Server program that could be given to ISP's and other large Internet service providers. This would allow the typical game player a large number of areas to meet with other players, but may also require a significant amount of support from FASA. This concept will be evaluated during the pre-production phase of the project.

Once connected to one of these services, the player may view all the current active game lobbies and their ping time to each of the lobby hosts. The player may then join any open lobby or private lobby that the player has password access to. In the event that the player doesn't find any games that they want to play in, they may start their own lobby.

The major differences between the lobbies in local vs. Internet games are how ping times are handled. In local games, ping times should be an entirely mute point. Latencies in excess of 30 are fairly rare on most local area networks. However, the Internet is rife with varying levels of ping times and the game lobby screen will indicate the quality of each players ping connection to the game host. This will be represented digitally and ranked via a simple color code. A high ping time will be red, medium will be yellow and low will be green. While we will design the game and networking systems to handle a



variable amount of latency, the best gaming experience will take place on low latency, low ping games. If a player has a very poor ping time it may be wise to find a different game to participate in.

PLAYER RANKING

Internet hosted games can optionally send summary updates to a player ranking server or perhaps a master server developed either by FASA as part of the FASA Master Server or to a ranking service provided by one of the game services.

The ranking server will track the top 1000 player scores and rank them based upon each player's actions in every game played online. These scores can then be viewed and compared by anyone who plays 'MechWarrior IV online.

Each player online will be given a specific user ID number that will remain active, so long as they play the game at least 1x per month. If a player does not log in at least one time a month with that user ID, the account will be automatically deleted.

Player ID scores will be tracked based on three different time scales - lifetime, monthly and weekly. Lifetime scores are fully persistent and remain active for the life span of the product. Monthly and weekly scores are reset at the end of each time period, respectively.

We will track three pieces of data for each player - Total Kills, Total Deaths and Total Damage Inflicted. Based on testing, we may expand statistic tracking to include damage inflicted on enemies, damage incurred, player accuracy %, etc.

Examples of potential ranking types follow:

Lifetime

Rank By:	Total Kills	User ID	Team	#
	1. Thorely	2341	Fasa	31093
	2. Paladin	234	Vweg	1934
	3. Jethro	32	'Mechman	1590
	...	...	...	...
	1000. Killer	39342	Freelance	39

Rank By:	Total Deaths	User ID	Team	#
	1. Killer	2341	Fasa	67
	2. Jethro	234	Vweg	645
	3. Paladin	32	'Mechman	1546
	...	...	...	...
	1000. Thor	39342	Freelance	3954

Rank By:	Kill Ratio	User ID	Team	#
	1. Thorely	2341	Fasa	7.8
	2. Jethro	32	'Mechman	2.4
	3. Paladin	234	Vweg	1.2
	...	...	...	...
	1000. Killer	39342	Freelance	0.58



Monthly

Monthly scores will be displayed just as lifetime scores, but will be ranked only by scores recorded within a one-month period. The player will also be able to list last month's top 100 in each category, purely for bragging rights.

Weekly

Monthly scores will be displayed just as lifetime scores, but will be ranked only by scores recorded within a one-week period. The player will also be able to list last week's top 100 in each category, purely for bragging rights.

NETWORK FEATURES

Spectator Mode

In addition to the basic eight players allowed in a network game, we allow an as of yet undetermined number of spectators who can view the multiplayer game as either a free - floating camera or as a camera attached to one of the player 'Mechs.

Dynamic Entry

So long as the host has left the lobby status, open players may join into games already in progress.

Custom Look

Players will be able to customize their individual appearance to other player's in a network game. A player will be able to set his personal skin type for the 'Mech, the team logo used on the 'Mech and also his pilot portrait used while chatting. Prior to joining a network game, these elements will be passed to all the other players in a single game lobby.



Scoring

The player will receive an experience score based on their actions in the game. Specific point values will be assigned to primary and secondary mission objectives, which they player will receive upon completion of each objective.

Bonus points will be gained by the player for taking our tertiary objectives, acquiring salvage goods and by weapons proficiency factor.

Negative points will be incurred based on losses taken in the mission, including a lost Pilot, 'Mechs or other vehicles.

Mission Tools

The game will ship with a mission creation tool, which will allow any player to generate completely unique levels. These new player levels can then be traded with other players online.

The mission creation tool will be a subset of the tool used by FASA for the creation of the actual levels that ship with the game.

With this tool, the player will be able to select a place object, edit object attributes and state, place AI vehicles and AI paths and set relationships between objects.

The one restriction the tool will enforce is the players will have to use worlds that were created by FASA. It will not be able to import new terrain meshes, maps or other artwork.

AI System

The ultimate goal of the AI system is to create intelligent group level responses to scripted or random situations.

Pilot AI's

The ultimate goal of the pilot AI system is to make each pilot have unique characteristics that appear human in nature. Each AI pilot will also increase in skill levels over time, which will be very apparent to the player.

Pilot AI Elements

Pilot AI's will consist of a few unique elements that will determine basic pilot personality characteristics. These and other basic element are path finding, target knowledge, target preference, weapons preference, range preference, heat management and targeting accuracy. Each of these specific elements will be driven by a code based AI. Other AI element layers such as Mission AI and Commander AI's will be script driven.

Target Knowledge

Each 'Mech Pilot will have specific technical information about each target in the game. This information is basically a table of how to best kill a specific target (as determined by the designers). This information will then be revealed to the Pilot AI based on



FASA INTERACTIVE

experience. If a pilot has never fought a Madcat, he will not have specific information and will rely on generic attack patterns during this first encounter. If the Pilot survives the encounter the next time he faces a Madcat he will “know” how to best kill it and modify his attack pattern accordingly. A pilot will also “know” the how to best kill an enemy ‘Mech if he has previously Piloted that ‘Mech model.

Through this system, a Pilot with more experience will become much more valuable to a player in the field. In contrast, a player with have much more of a challenge when facing an experienced Pilot.

Target Preference

Target preference is what specific area the Pilot will target. This selection will be based upon available target knowledge, orientation to the target, range to the target and prior target damage.

If the Pilot knows the back of a Madcat is “soft” it will choose that location as the target preference and maneuver to get behind the enemy ‘Mech.

Weapons Preference

Each Pilot will have a particular weapons preference as a character default. Based on the current target, the to the target the Pilot will choose an appropriate weapon to fire.

Range Preference

This is a critical element of each Pilot’s character. A super aggressive in your face Pilot will like short-range weapons, while a cool methodical Pilot might prefer a longer-range weapon. When faced with an enemy target, the Pilot will always attempt to move to his preferred range.



Heat Management

Each Pilot will have a predetermined heat plan that he will follow. This plan will determine how intense battles with each Pilot can become.

A cool methodical Pilot will never allow his heat to reach critical levels.

A dangerous wild Pilot will completely unload his weapons always pushing the edge of meltdown and if he thinks he can perform a quick kill on an enemy, the Pilot will push the heat level to the max and risk his own death for the kill.

Targeting Accuracy

Based on experience a Pilot AI will increase in overall targeting accuracy %. As the computer will know exactly where to shoot to score a hit, we will reduce this accuracy by introducing a random accurate targeting variable that will change relative to the Pilots' experience.

Pilot / Commander AI Interaction

Another critical element of Pilot AI is how they “know” what to do in a given situation. The Pilot AI itself is rather “dumb” and will only make decisions based on higher level commands given via a higher AI level known as the Commander AI.

A Commander AI will order a pilot to “guard”; the Pilot will know how to guard but will only do this once it has received this order. If an enemy then attacks an area that the Pilot is not guarding, the Pilot will not respond unless it is ordered by the Commander AI to do so. If three ‘Mechs are on a guard patrol and encounter two enemy ‘Mechs the Commander AI will determine which guard ‘Mechs engage which enemy ‘Mech.

Commander AI's can also pass specific information to a Pilot AI which will modify it's specific AI bases, such as Patrol With Active Sensors, etc.

By separating the Pilot AI and Commander AI elements we can create an intelligent response system wherein numerous vehicles can respond to random situations in a coordinated effort.

The Commander AI

The commander AI controls groups of pilots. The idea of the commander is an evolution of “Nanny” from saucer. In code it is much easier to deal with multiple AI communication as a one way street, or I tell you what to do and you do it. The pilots get to choose how they go about their orders, but they will always follow their orders to the death. To make sure that pilots don't die the Commander AI is responsible for pulling the pilot out of battle or giving the pilot. From a designing standpoint the only difference is the ability to have completely separate groups of Mechs that are autonomous. For the most part you are assigning “Pilots” instructions, but a Commander AI is taking care of the details in script. Since Commander AI's and Mission AI's share flags, scripted communication and interaction can happen.

Example Commander AI for above scenario:

[Kurita Loki 1]
mech=Kurita Loki 1



FASA INTERACTIVE

```
Guard="Kurita WaterTower"; radius 20 meters; // this guards the water tower
Stay In Radius="Kurita Water Tower"; radius 20.0 // this won't let the pilot chase farther than 20
meters
```

```
[Kurita Loki 2]
mech=Kurita Loki 2
Guard="Kurita WaterTower"; radius 20 meters; // this guards the water tower
Stay In Radius="Kurita Water Tower"; radius 20.0 // this won't let the pilot chase farther than 20
meters
Selflove=0 // this says never run away. The same as not setting it!
```

```
[Kurita Loki 3]
mech=Kurita Loki 3
protect="Kurita Loki 1"; priority .5 // this makes 3 protect 2 and 1 an equal amount
protect="Kurita Loki 2"; priority .5 // options can be extended to include more concrete
// data about damage.
Selflove=.2 // tells this mech to run away at 20% damage
```

```
[Kurita Thor 4]
mech=Kurita Thor 4
if set="Flood"
GoTo="Exit Point"; speed=1.0 // this tells it to flee at full speed.
```

```
// Commander rules can also be made, this is just kind of a second way to do things.
// It should help protect against the unknowns.
```

```
[Commander Rule 1]
Priority=100
Commander Protect="Kurita HQ"
UseClosestMech="Kurita Thor 1"; "Kurita Thor 2"; "Kurita Thor 3"; "Kurita Thor 4";
```

```
// We use a priority setting to make sure that the HQ gets protected before gun turret
[Commander Rule 1]
Priority=20
Commander Protect="Kurita Gun Turret"
UseClosestMech="Kurita Thor 1"; "Kurita Thor 2"; "Kurita Thor 3"; "Kurita Thor 4";
```

More work on mission design needs to be done so we can see where we need to go. We should “simulate” making a few missions and try making rules for them. More fleshing out of the keywords, priority system, threat assessment, AI Focus (it doesn’t change it’s mind every frame), need to be done for the AI design.

Mission AI

The Mission AI keeps track of global events, win conditions, all AI’s, and scores and is mostly if not entirely script driven.

Score

Scoring will have code set up for each of the different types of scoring methods. Each scoring method will be customizable so that you can potentially have variations of the same scoring method. Example would be a Team vs. Team with no friendly fire penalty.



Suggested models:

Single Player

Campaign
Instant Action

Multiplayer

Team vs. Team
Free For All
Deathmatch
King of the Hill
Capture the flag
Team Defend Attack

Scripting Win Conditions

Win conditions can be cascaded and there can be as many branches as needed. The cascades must all return true for the win condition to be met. In addition rules can be marked as return true once.

Example Win Condition

The player must destroy Loki 1 before 10:00. Secondary Objective: Player must wait until the board meeting at 15:00 to destroy the Head Quarters. Then the player must go to the drop zone.

Example Of Format

[Primary 1 Rules]
if destroy ="Kurita Loki 1"
if timer before=10:00
set="primary1"

[Secondary 2 Rules]
if destroy ="Kurita HQ"
if timerafter=15:00
set="secondary 1"

[DustOff Rules]
if position="DustZone" // coment: DustZone is a rectilinear region in the map
dustoff

Scripting Events:

Events can be scripted and responded to with these rules. Flags can be set and checked. This way more complex behavior can be scripted.

Example:

[Primary 1 Rules]
if destroy ="Kurita Loki 1"



FASA INTERACTIVE

© Copyright 1998 FASA Interactive Technologies, Inc.
MechWarrior is a trademark of FASA Corporation.
Confidential – For Internal Use Only – DO NOT DISTRIBUTE

```
if timer before=10:00  
set="primary1"
```

```
[Secondary 2 Rules]  
if destroy ="Kurita WaterTower"  
set event="Flood" // set the script flag  
do spwan=Entity::Water ; point="Kurita WaterTower" // make the effect at the water  
tower
```

```
[Cruze Line Rules]  
if position="Boat Dock" // coment: Boat Dock is a rectilinear region in the map  
if set="Flood" // make sure we are flooded  
dustoff
```